# A Collaborative Filtering Probabilistic Approach for Recommendation to Large Homogeneous and Automatically Detected Groups

Remigio Hurtado[1,2]*, Jesús Bobadilla[2], Abraham Gutiérrez[2], Santiago Alonso[2]

[1] Universidad Politécnica Salesiana, Calle Vieja 12-30, 010102 Cuenca (Ecuador)
[2] Universidad Politécnica de Madrid, Carretera de Valencia Km 7, 28031 Madrid (Spain)

UNIR
LA UNIVERSIDAD
EN INTERNET

## Abstract

In the collaborative filtering recommender systems (CFRS) field, recommendation to group of users is mainly focused on stablished, occasional or random groups. These groups have a little number of users: relatives, friends, colleagues, etc. Our proposal deals with large numbers of automatically detected groups. Marketing and electronic commerce are typical targets of large homogenous groups. Large groups present a major difficulty in terms of automatically achieving homogeneity, equilibrated size and accurate recommendations. We provide a method that combines diverse machine learning algorithms in an original way: homogeneous groups are detected by means of a clustering based on hidden factors instead of ratings. Predictions are made using a virtual user model, and virtual users are obtained by performing a hidden factors aggregation. Additionally, this paper selects the most appropriate dimensionality reduction for the explained RS aim. We conduct a set of experiments to catch the maximum cumulative deviation of the ratings information. Results show an improvement on recommendations made to large homogeneous groups. It is also shown the desirability of designing specific methods and algorithms to deal with automatically detected groups.

## I. Introduction

THIS section is divided into four subsections: 1) Fundamental concepts of Recommender Systems (RS), 2) Clustering to improve collaborative filtering, 3) Matrix decomposition-based clustering, and 4) Recommendation to groups of users and the proposed approach.

### A. Recommendations to Individual Users

RS [1] field is relevant in the Artificial Intelligence scenario since it significantly reduces the information overload problem on the Internet. RS suggest to the users about the items they probably will like. Depending on the item nature, a variety of RS can be implemented: e-learning [2], tourism [3], [4], films [5], restaurants [6], networks [7], healthcare [8], industrial operators [9], etc. The most accurate type of RS is the Collaborative Filtering (CF) one [10],[11]. CF RS are based on the preferences of users about items; preferences can be explicit (votes) or implicit (listened songs, purchased items, watched movies, etc.). CF RS have been traditionally implemented by using the K-Nearest Neighbors (KNN) machine learning method [12], although current CF RS kernels are usually based on the Matrix Factorization (MF) algorithm [13], [14]. MF converts the sparse matrix of ratings (users x items) to two dense matrices: 'users x factors' and 'factors x items'. Since the number of factors is very little (usually 20 to 80),

MF makes a matrix reduction [15] and it extracts the most important information from the original sparse rating matrix. Factors are called 'hidden factors' and they condense the relevant information from the CF dataset. Predictions to each user can be computed by making the dot product of the user and the items factors. Well known MF variations are Positive Matrix Factorization (PMF) [16], [17], Bayesian Nonnegative Matrix Factorization (BNMF) [18], and Elementwise Alternating Least Squares (eALS) [19]. Finally, state of the art CF RS implementations make use of neural networks [76], some of them joining MF and the Multilayer Perceptron [20]. These state of the art neural network approaches cover many different scopes, such as music [21] and videos [19]. Beyond CF, it is usual to improve accuracy results by merging RS types (hybrid RS) [22]. CF RS can be reinforced by means of demographic [6], content-based [23], context-aware [24], and social [7] information.

### B. Clustering to Improve Collaborative Filtering

Clustering is a recurrent resource to improve CF RS [25]. Clustering can be performed on several types of RS information: content-based [26], demographic [27], hybrid [28], etc. CF clustering has traditionally performed on items [29] or users [30]; CF clustering based on the ratings information is the current most published approach [31], [25]. Fuzzy C-Means has proved to be accurate to CF RS purposes [26] and to improve coverage. MF can also be improved by means of a user clustering model [32]. To establish the number of clusters parameter (K) is a trial and error process that requires knowledge of the data and experience; in [33] the number of clusters can be dynamically set

* Corresponding author.
E-mail address: rhurtadoo@ups.edu.ec

to arrange the RS data size variations, and results show an accuracy improvement. The co-clustering method has also been used to improve MF results [34]. Making use of the user changes on their preferences, an evolutionary clustering algorithm has been proposed [35]; it takes into account the temporal evolution of features. Clustering similar items is a simple and useful way to improve CF RS accuracy; this is done with e-commerce products on [36]. Learning analytics with clustering is a potential benefit that grows with the size of the users. This has improved digital education processes on [37]. Association rules mining has been used to reduce the size of the data, and then to make the clustering process more efficient. Specifically, [38] has reduced the item space. Centroid selection is a key process to tackle most of the CF clustering approaches; it improves performance and reduces the processing time [39].

### C. Matrix Decomposition-Based Clustering

In the context of this paper, it is important to address the matrix decomposition problem for clustering purposes. In the CF field, the use of MF provides some relevant clustering advantages [40]: 1) MF accurately models sparse data variations [41], 2) It can implement both hard and soft clustering: e.g.: by means of Nonnegative Matrix Factorization (NMF) and BNMF, and 3) MF simultaneously factorizes users and items. Co-clustering is a clustering approach used in CF; it can be used to discover space correlations in big data scenarios [42]. An NMF-based semi-supervised co-clustering is proposed in [43] involving CF link relations. When CF datasets incorporate constraint restrictions, performance can be improved by means of clustering approaches. The regularized NMF incorporates constraints support to the standard NMF method, such as neighborhood-based local learning regularization [44]. The constraint restrictions methods rely on datasets that incorporate this type of information: they cannot be considered generalized, such as the one we propose in this paper.

### D. Recommendation to Groups Of Users and Proposed Approach

Once the above three blocks (CF RS, clustering and MF-based clustering) have been addressed we will focus on the CF recommendations to groups of users. The recommendations to groups of users arise from the convenience of being able to recommend a group of users about products or services that satisfy the entire group [45].

Group RS can be classified according to the group type [46]: a) *Established group*: persons that belong to a stable group; b) **Occasional group**: persons that at times join to a group, c) *Random group*: persons who share an environment in a particular moment, and d) *Automatically identified group*: automatically detected groups, considering the preferences of users. From [47] we can distinguish between homogeneous and heterogeneous groups. **Homogeneous groups** are established by the System, whereas **heterogeneous groups** are dynamically created by the users. Most of the research has focused on established and heterogeneous groups [48], [49], [50], [51]. Our proposed approach is designed to make recommendations on homogeneous and automatically identified groups [46], [52], [53]. Homogeneous groups are particularly relevant for marketing processes, where companies want to recommend products or services to a broad target of similar users. Homogeneous groups are usually obtained by using non-supervised machine learning methods, such as diverse clustering approaches [5].

The **main objective** of this paper is to improve the quality of recommendations made to automatically detected homogeneous groups in a RS. With this purpose, it is essential to divide the RS set of users in the very best homogeneous groups. Our *first hypothesis* is that we can improve the detection of homogeneous groups by performing a clustering that combines the appropriate MF dimensionality reduction

with the aggregation of factors. The **second hypothesis** is that the RS recommendation quality will be improved by combining the proposed aggregation model with the designed clustering approach, both based on factors. Our probabilistic approach aimed at groups of users allows predicting the probability that a virtual user (group) likes a specific item.

The **proposed dimensionality reduction** is based on BNMF method [18]. We have made experiments to compare BNMF, Principal Component Analysis (PCA), and TruncatedSVD dimensionality reduction techniques. Since PCA computes the covariance matrix, it operates on the entire sparse matrix, whereas BNMF and Truncated Singular Value Decomposition (TruncatedSVD) do not have this limitation. The obtained results show that to recommend homogeneous groups, BNMF outperforms PCA and TruncatedSVD: 1) It obtains better variance values than the baselines, so it needs fewer dimensions (factors) to provide the same information level, 2) It provides better within-cluster results and also a more equilibrated number of users in its clusters, and 3) BNMF returns factors with probabilistic meaning that can be used in the prediction stage. As we will see later, the BNMF hidden factors will be used both to feed the clustering process and to make virtual user models by aggregating factors of users.

In Fig. 1 we explain the *current research context of the proposed approach*. This figure shows five states of the art methods, from a) to e), designed to make recommendations to groups of users. The method labeled as f) corresponds to the proposed one in this paper. Method a): **Recommendation fusion** [49] is known as RANK; it makes an aggregation of the set of individual recommendation made to the users of the group. Method b): **Prediction aggregation** [54] is known as PER an it makes an aggregation of the set of individual predictions made to the users of the group. Method c): **User preferences aggregation** makes a model of the group of users (virtual user); this is a synthetic user that represents to whole set of users in the group. Once the virtual user is obtained, the traditional recommendation to one user process is done; this approach is the more accurate one when applied to heterogeneous groups. We call it VUR (*Virtual User based Recommendation*) [45], [51], [55] and we use it as a baseline. Method d) [56], [52] makes predictions before clustering, it performs aggregation post-clustering and it does not use dimensionality reduction. We will use it as a baseline using the name PC (*Predict & Cluster*). Method e): this is a variant of the method "b". We call it as RAP (**Recommendation via Aggregation of Predictions**) [57]; using RAP, dimensionality reduction is made before clustering, and predictions are obtained from ratings. We will use it as baseline to test the dimensionality reduction impact on the results.

The proposed method, called RAF (Reduction and Aggregation of Factors) (Fig. 1f) adds a dimensionality reduction that is not included in methods a) to d). Both the proposed method and the RAP one (Fig. 1e) perform the clustering by using the obtained hidden factors in the dimensionality reduction process. Methods a) to b) run the clustering algorithm by using the dataset ratings information. Since both the baseline RAP and the proposed RAF methods make the same clustering, their homogenous groups are not different. The crucial differences between the proposed RAF and the baseline RAP methods are: 1) RAP characterizes users by their preferences (ratings), whereas RAF characterizes users by their hidden factors; consequently, RAF provides a higher semantic level than RAP, and 2) RAP first makes individual predictions and then aggregates them, whereas RAF creates a model (virtual user of the group) and then obtains its prediction. It is relevant to highlight that the RAF virtual user is an aggregation of hidden factors, whereas the RAP prediction, is an aggregation of user predictions based on ratings.

The rest of the paper is divided into the following sections (with the same numbering shown here):

II. Related work, in which a review is made of the most relevant contributions that exist in the CF aspects covered in the paper.
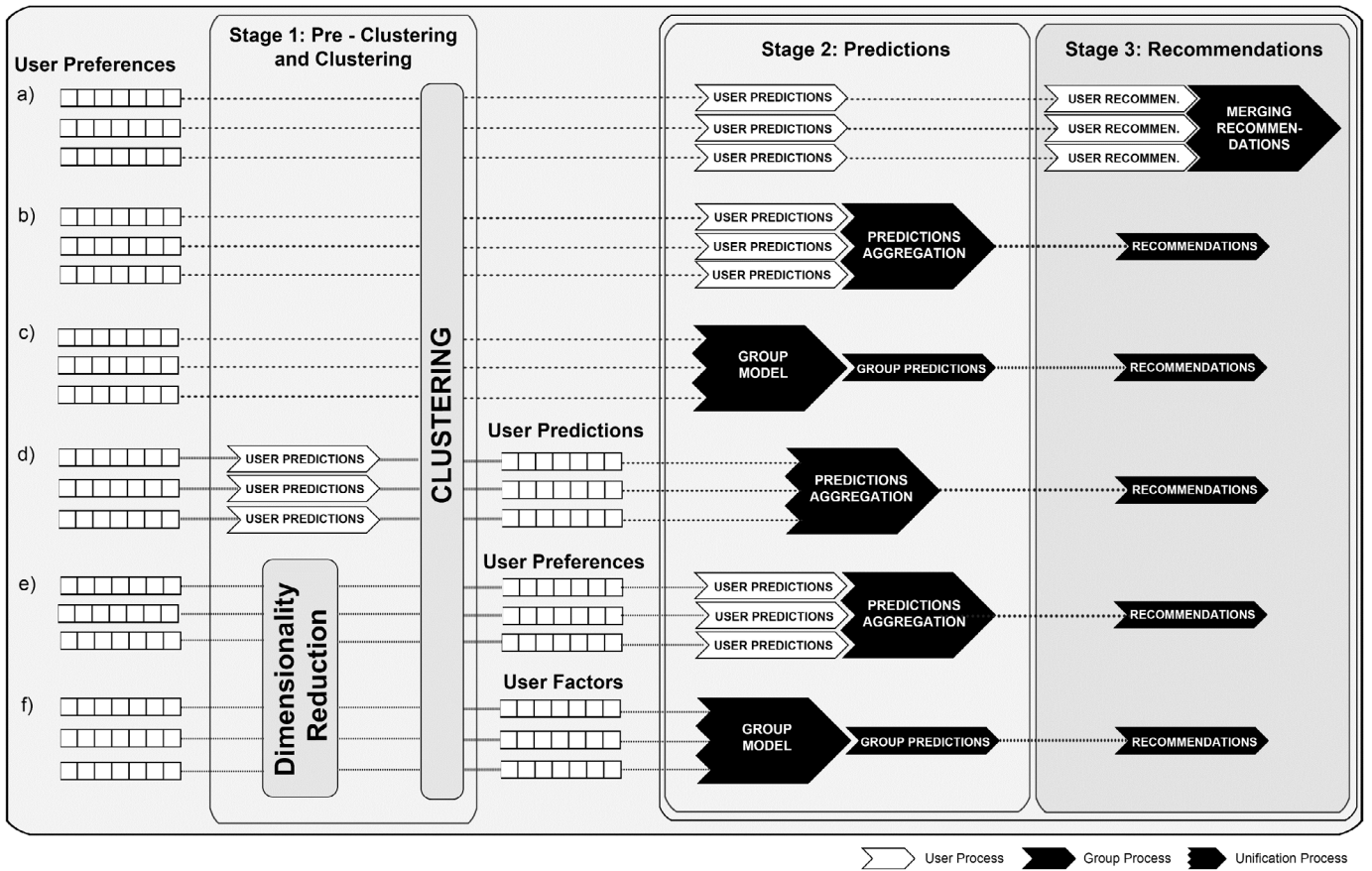
Fig. 1. State of the art approaches for recommendation to groups of users.

III. Formalization of the proposed CF method which specifies the way to predict and recommend. In addition, we provide a running example.

IV. Experiments and results: experiments set up, quality measures, model optimization of parameters, model performance, comparative results, and discussion.

V. Most relevant conclusions obtained and future works.

## II. RELATED WORK

In Fig. 1, from methods a) to d) we show the most relevant researches in the context of the proposed approach (designed to make recommendations to groups of users). This works are: a) Recommendation merging [49], b) Prediction aggregation [54], c) User preferences aggregation [45], [51], [55], and, d) Predict & Cluster [56], [52]. In addition to these methods, there are other models and methods based on matrix factorization, neural collaborative filtering, clustering methods, and graph approaches. From these methods we can highlight the following researches:

Virtual users can be extracted from MF factors [50] to recommend to each group of users by means of its corresponding group virtual user. Authors perform their experiments using groups of sizes: 2 to 4 users, 5 to 8 users, 9 to 12 users. For the state-of-the-art papers in the field, this is a typical range of user size in a group. Conversely, this paper makes use of much larger numbers of users in each group. Design alternatives for CF recommendation to a group of users are tested in [51]; conclusions point to a relevant increase of performance (execution time) when aggregation is made in early stages, whereas accuracy does not significantly change according to the stage where the aggregation is made. Our paper borrows this concept to design the

proposed approach. A model based on the topic of argumentation [58] is used to recommend personalized items for groups. The argumentation subject is extracted and the users with similar views are clustered into groups. Uncertainty is used in [59] to model the way members might agree on a group ranking. Based on the observed member's individual rankings, they quantify the likelihood of group rankings.

To use embedding representations of CF data contributes to improving results. A Neural Collaborative Filtering design is presented in [60] where a neural network learns the interactions of groups and items; it uses factor embeddings. On the same line [61] uses group activity history information and recommender post-rating feedback to generate interactive preference parameters. In the same line that our proposal, [62] combines latent factor models to obtain improved group recommendations. In this case, a multi-layer perceptron is used to make this task. State of art in recommendation to groups of users includes a broad range of application fields: authors in [63] use a dragonfly algorithm to deal with sparsity; they provide a client-based collaborative filtering approach and apply it to restaurant recommendation. Considering repeat purchasing, [64] provides a group RS to optimize the offline physical store inventories. A large study with real groups of tourists [65] manages the problem of finding a sequence of points of interest that satisfies all group members. A travel RS for individual and group of users is also proposed in [66]; this RS provides a list of POI as recommendations. They also exploit relationships between users. Recommendation of clinics to patient groups is provided in [67], where it divides patients into multiple groups by mining their unknown preferences before recommending them suitable clinics. An academic venue RS is proposed [68], where academic venues are recommended for a group of researchers based on the venues attended by their co-authors, the group members and also on their co-citers. Group cohesion is a key factor in group

recommendation, and clustering is a powerful tool to reach cohesion. Heterogeneous information networks can contain rich information about entities and relationships: [69] provides an approach for group recommendation that appropriately captures group cohesion. Social information can also improve group recommendation results: To detect the inherent associations among group members, [70] incorporates user social networks into the random walk with restart model.

Recommendations explanation is a difficult task, particularly in group recommendations; [71] investigates which explanation best helps to increase the satisfaction of group members, to improve fairness perception and to obtain consensus perception. Visualization of group recommendations is also a challenging task; authors in [72] provide visual presentations and intuitive explanations. Finally, using social trust information, [73] identifies trustworthy users and it analyses the degrees of trust among users in a group.

## III. Proposed Method

This section introduces the architecture of the proposed method and its relevant details. Explanations are reinforced by means of a data toy running example that helps to understand some internal functionalities. Fig. 2 shows the architecture of the method that we will use as a roadmap for the section explanations. From Fig. 2 we find the following procedural blocks:

1. Our starting point is the CF set of ratings: We are not proposing a hybrid RS [28]; we provide a pure CF RS method that can be used in all types of CF datasets.

2. The second block in Fig. 2 implements the currently most used model to obtain individual recommendations on RS: Matrix Factorization. We have selected the Bayesian Non-Negative Matrix Factorization (BNMF) [18], since we have made experiments (explained in the next section) and it has been found that BNMF outperforms PCA and TruncatedSVD in several aspects: mainly, it obtains better variance values and it provides both better within-cluster results and a more equilibrated number of users in their clusters. The BNMF mathematical formulation can be found in [18]; it is out of the scope of this paper.

3. Machine learning clustering methods return better results when the iterative algorithms are fed with appropriate initial values. Pre-clustering is a very dependent task on the data scope. RS clustering can be particularly improved by using power users [13], [39], and then we make this process. The algorithm *KMeansPlusLogPower* chooses *K* users from the dataset, as centroids. Its high-level algorithm is:

---

**Algorithm:** KMeansPlusLogPower

**Input:** *U*, hidden factors of the users in training set; *k*, total number of clusters

**Output:** *k* centroids, {*c1, c2, · · · ck*}

---

1: Define desired number of clusters, *k*

2: Select the initial centroid *c1* to be *Pu*

3: **repeat**

4:     Select the next centroid ci where *ci = u' ∈ U* with the probability: *Prob = distance(u) + log[ (1/p(x)) + 1]*

5: **until** *k* centroids are found

6: **return** {*c1, c2, · · · ck*} "*k* centroids"

---

*p(x)* is the relation between the number of items consumed by *u'* user and the number of items consumed by the *Pu* power user.

4. The proposed method runs the clustering algorithm by processing the user hidden factors (and not the user ratings). In this paper, we conclude that hidden factors provide better quality results than ratings. The reasons behind the result are: a) They offer a higher level of abstraction, b) Its dimensionality is much lower than the ratings one, c) They are not sparse.

Clustering is made using the set of hidden factors from each RS user. It can be defined as:

Let $U$ be the set of users $\qquad\qquad(1)$

Let $I$ be the set of items $\qquad\qquad(2)$

Let $F$ be the set of Hidden Factors $\qquad\qquad(3)$

Let $f_j$ be the hidden factors of user j

$$f_j = \{f_{j,1}, f_{j,2}, f_{j,3}, \dots, f_{j,F}\} \mid j \in [1..U] \qquad(4)$$

5. The proposed method makes a model where each class is represented by a virtual user [50]. The virtual user is obtained by aggregating the hidden factors of all the users belonging to the class. The key concept here is that the virtual user accurately represents the users of its class because its limited number of factors contains the most relevant information. Aggregating ratings (instead of factors) cannot catch the relevant information and it does not provide a representative virtual user.

Let $K$ be the number of clusters $\qquad\qquad(5)$

Let $u_j$ be the user j $\mid j \in [1..U]$ $\qquad\qquad(6)$

Let $v_k$ be the virtual user k $\mid k \in [1..K]$ $\qquad\qquad(7)$

Let $C_k$ be the set of users in the cluster k $\qquad\qquad(8)$

Let $f_{v_k,f}$ be the factor f of the virtual user k

$$f_{v_k,f} = \frac{1}{\#C_k}\sum_{j \in C_k} f_{j,f} \qquad(9)$$

Let $f_{v_k}$ be the hidden factors of virtual user k

$$f_{v_k} = \{f_{v_k,1}, f_{v_k,2}, f_{v_k,3}, \dots, f_{v_k,F}\} \mid v_k \in [1..K] \qquad(10)$$

6. To make predictions for each group (class), the proposed method is based on the BNMF individual predictions schema [18]. In this case, our method contains two parameters:

- $\alpha$, which controls the amount of overlapping of a user between user factors.

- $\beta$, that fixes the amount of evidence needed to determine that an item factor is associated with a factor of a virtual user.

The association between item factors and virtual user factors allows obtaining the probability that a virtual user $v_k$ likes a specific item $i$. This probability is obtained through the dot product of the virtual user factors and the set of item factors. Subsequently, this probability is transformed to the rating scale of the dataset, and the virtual user prediction to the item $i$ is obtained, this is $p_{v_k,i}$.

Let $f'_i$ be the hidden factors of item $i$

$$f'_i = \left\{f'_{i,1}, f'_{i,2}, f'_{i,3}, \dots, f'_{i,F}\right\} \mid i \in \left[1..I\right] \qquad(11)$$

Let $p_{v_k,i}$ be the prediction made to the virtual user $k$ on item $i$

$$p_{v_k,i} = \sum_{f \in F} f_{v_k,f}\, f'_{i,f} \mid k \in \left[1..K\right], i \in \left[1..I\right] \qquad(12)$$
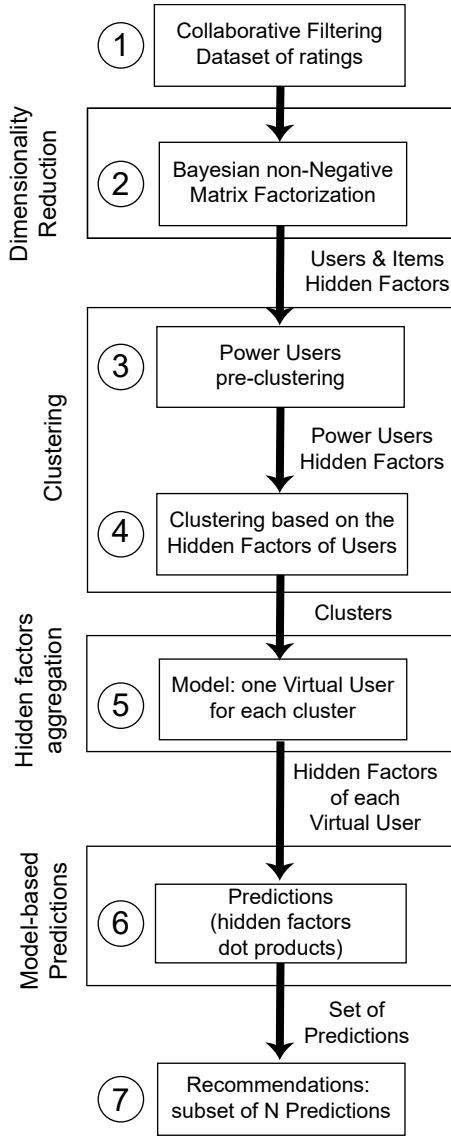
Fig. 2. Proposed method architecture.

In summary, this block of predictions is a probabilistic Bayesian approach of recommendation for groups of users. This approach allows us to compute:

- The probability that a virtual user likes a specific item.
- The prediction for the virtual user (group). The underlying idea is to predict to the set of users of a cluster the same predicted items for their common virtual user.

7. Recommendations, as usual, are selected from the $N$ highest predictions [1].

Let $R_k$ be the set of recommendations made to the class $k$ and $\#R_k = N$

$$R_k = \left\{ i \in I \mid p_{v_k,i} \geq p_{v_k,j} \ \forall i \in R_k, \forall j \in R_k^c \right\} \quad (13)$$

Previously to the explained method, the BNMF factorization approach has been selected to feed the clustering algorithm. The next section includes experimental results that show the BNMF is able to hold more information than PCA and TruncatedSVD, by using the same number of factors. Experiments have been run using RS data. The BNMF accumulated variance is obtained implementing (14) to (17).

Let $\sigma_f$ be the set of factors variances

$$\sigma_f = \left\{ \sigma(f_1), \sigma(f_2), \sigma(f_3), \dots, \sigma(f_F) \right\} \quad (14)$$

Let $\sigma(f_f)$ be the factor f variance

$$\sigma(f_f) = \frac{\sum_{u \in U}(f_{u,f} - \bar{f}_f)}{\#U} \quad (15)$$

Let $\theta$ be the threshold of required accumulated variance $\quad (16)$

Let T be the set of factors that hold the required accumulated variance

$$T = \{i \in F \mid \sigma(f_i) \geq \sigma(f_j) \ \forall i \in T, \forall j \in T^c, \sum_{i \in T} \sigma(f_i) \geq \theta\} \quad (17)$$

To fix the proposed method concepts we provide a data toy example. Fig. 3.1 contains the CF dataset ratings casted for 10 users ($u_1$ to $u_{10}$) on 15 items ($i_1$ to $i_{15}$). To run the BNMF method we have chosen $F=3$. Fig. 3.2 shows the obtained factors for each user (on the right of the dataset) and the obtained factors for each item (on the bottom of the dataset). The clustering process has been run using a $K=5$; the obtained clusters (groups) are shown in Fig. 3.3. Fig. 3.4 contains the factors of each one of the five virtual users representing the five groups. The factor values are obtained by implementing (5) to (10). Making the dot product of each virtual user factors and each item factors predictions are obtained (Fig. 3.5): (11) and (12). Finally, from predictions we can obtain each group recommendations; e.g.: using $N=3$, group 1 recommendations are: $i_1$, $i_2$, $i_3$; group 2 recommendations are: $i_6$, $i_7$, $i_9$; ... group 5 recommendations are: $i_{10}$, $i_{12}$, $i_{15}$.

## IV. Experiments and Results

This section explains the design of the experiments: chosen datasets, quality measures, parameter values, etc., and their results. Experiments have been divided into two phases: (1) finding the optimal parameters of the proposed method to each tested dataset, (2) comparing the proposed method with state of art to make CF recommendations to homogeneous groups of users. All the experiments have been carried out using public datasets widely used by RS research papers. We have selected MovieLens [74] and FilmTrust [75] datasets. Table I contains the most relevant facts about these datasets. Finally, we test the *RMSE* prediction quality measure and the *F1* recommendation quality measure. Cross-validation values used in the experiments are abstracted in Table II.

### A. Experiments Set Up

To perform the experiments, we have split users and items into test and training sets. To avoid fluctuations, we perform each experiment using 10-folds Monte Carlo cross-validation. Table I and Table II contain the main parameters of each dataset, chosen for the execution of experiments.

TABLE I. Main Properties of the Datasets Used in the Experiments

| Dataset | Number of ratings | Number of items | Number of users | Rating values |
|---|---|---|---|---|
| MovieLens | 1,000,209 | 3,706 | 6,040 | 5-star scale, with half-star increments |
| FilmTrust | 33,470 | 2,059 | 1,227 | 0.5 to 4 with half increments |

**1. Rating matrix**

| $r_{ui}$ | $i_1$ | $i_2$ | $i_3$ | $i_4$ | $i_5$ | $i_6$ | $i_7$ | $i_8$ | $i_9$ | $i_{10}$ | $i_{11}$ | $i_{12}$ | $i_{13}$ | $i_{14}$ | $i_{15}$ | $F_1$ | $F_2$ | $F_3$ | idGroup |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $u_1$ | 5 | 5 | | 3 | | | 3 | | | 2 | | | 1 | | | 0.1 | 0.1 | 0.81 | 1 |
| $u_2$ | 5 | 5 | 5 | 4 | | | | | | 1 | | | 1 | | | 0.1 | 0.1 | 0.81 | 1 |
| $u_3$ | | 2 | | 5 | 5 | 5 | | | | 3 | | | | | | 0.77 | 0.12 | 0.11 | 2 |
| $u_4$ | | | | 5 | 5 | 5 | | | | 5 | | | 3 | | | 0.73 | 0.17 | 0.1 | 2 |
| $u_5$ | | | 2 | | | | 5 | 5 | 5 | | 2 | | | 2 | | 0.67 | 0.12 | 0.21 | 3 |
| $u_6$ | | | 1 | | | | 5 | 5 | 5 | | | | | 1 | | 0.66 | 0.11 | 0.23 | 3 |
| $u_7$ | | | | | | | | 1 | | | 5 | 5 | 5 | 3 | | 0.12 | 0.75 | 0.13 | 4 |
| $u_8$ | | 1 | | | 2 | | | 2 | | | 5 | 5 | 5 | 2 | | 0.13 | 0.76 | 0.12 | 4 |
| $u_9$ | | 2 | | 1 | | | | | | | 2 | | 5 | 5 | 5 | 0.19 | 0.69 | 0.13 | 5 |
| $u_{10}$ | | | | 2 | | | | | | | | 1 | 5 | 5 | 5 | 0.23 | 0.66 | 0.11 | 5 |

**2. Item factors**

| | $i_1$ | $i_2$ | $i_3$ | $i_4$ | $i_5$ | $i_6$ | $i_7$ | $i_8$ | $i_9$ | $i_{10}$ | $i_{11}$ | $i_{12}$ | $i_{13}$ | $i_{14}$ | $i_{15}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $F_1$ | 0.51 | 0.3 | 0.5 | 0.7 | 0.8 | 0.8 | 0.8 | 0.8 | 0.8 | 0.6 | 0.2 | 0.5 | 0.5 | 0.3 | 0.5 |
| $F_2$ | 0.51 | 0.2 | 0.5 | 0.5 | 0.3 | 0.5 | 0.5 | 0.3 | 0.5 | 0.8 | 0.7 | 0.8 | 0.8 | 0.7 | 0.8 |
| $F_3$ | 0.78 | 0.8 | 0.8 | 0.3 | 0.6 | 0.5 | 0.5 | 0.4 | 0.5 | 0.3 | 0.4 | 0.5 | 0.2 | 0.5 | 0.5 |

**5. Group predictions**

| | | | | | | | | | | | | | | | | **4. Aggregation** | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $P_{G1}$ | 4.65 | 4.4 | 4.6 | 2.8 | 3.8 | 3.7 | 3.7 | 3.3 | 3.7 | 2.9 | 3.2 | 3.7 | 2.5 | 3.3 | 3.7 | 0.1 | 0.1 | 0.81 |
| $P_{G2}$ | 3.68 | 2.9 | 3.7 | 4.3 | 4.4 | 4.6 | 4.6 | 4.4 | 4.6 | 4 | 2.6 | 3.7 | 3.6 | 2.8 | 3.8 | 0.75 | 0.14 | 0.11 |
| $P_{G3}$ | 3.83 | 3.1 | 3.8 | 4.1 | 4.4 | 4.5 | 4.5 | 4.2 | 4.5 | 3.8 | 2.7 | 3.7 | 3.4 | 2.9 | 3.7 | 0.66 | 0.12 | 0.22 |
| $P_{G4}$ | 3.7 | 2.6 | 3.7 | 3.4 | 2.9 | 3.7 | 3.7 | 2.9 | 3.7 | 4.6 | 4.2 | 4.6 | 4.4 | 4 | 4.6 | 0.13 | 0.75 | 0.12 |
| $P_{G5}$ | 3.7 | 2.6 | 3.7 | 3.5 | 3.1 | 3.8 | 3.8 | 3 | 3.8 | 4.5 | 4 | 4.5 | 4.3 | 3.8 | 4.5 | 0.21 | 0.67 | 0.12 |

Fig. 3. Data toy to explain the proposed method.

TABLE II. Cross-validation Values used in the Experiments

| Parameter | Values |
|---|---|
| Testing-Ratings | 20% |
| Training-Ratings | 80% |
| #clusters ($K$) | MovieLens: {20 to 200 step 5} FilmTrust: {20 to 200 step 5} |
| #recommendations ($N$) | 15 |

## B. Quality Measures

The quality measures we use in the experiments are:

- *Root Mean Square Error (RMSE)*: it is a collaborative filtering prediction quality measure. The *RMSE* can be formalized as:

$$RMSE = \frac{\sum_{i \in I}(r_{u,i} - P_{G,i})^2}{n} \tag{18}$$

Where $r_{u,i}$ is the user $u$ rating to the item $i$. $P_{G,i}$ is the prediction for the item $i$ for the group $G$ in which user $u$ is, hence $P_{G,i} = p_{v_k,i}$. $I$ is the set of all items and $n$ is the number of ratings available in the test set. Low *RMSE* values are better since it means that prediction errors are lower.

- *F1*: in order to evaluate the quality of recommendations to groups of users, we define *F1* as the harmonic mean that combines the values of precision and recall. We define precision and recall for group *G*, as:

$$precision_G = \frac{\sum_{u \in G} precision_u}{\#u} \tag{19}$$

$$recall_G = \frac{\sum_{u \in G} recall_u}{\#u} \tag{20}$$

Where $G$ is the group in which user $u$ is, $precision_u$ and $recall_u$ is the precision and recall for the user $u$, respectively. The $precision_u$ and $recall_u$ can be formalized as:

$$precision_u = \frac{\#TP}{\#(TP \cup FP)} \tag{21}$$

$$recall_u = \frac{\#TP}{\#T} \tag{22}$$

Where *TP*, *FP*, and *T* denote the true positive, false positive and expected recommendations sets, respectively:

$$FP = \{i \in L_G \mid r_{u,i} < \theta \wedge r_{u,i} \neq \bullet\} \tag{23}$$

$$TP = \{i \in L_G \mid r_{u,i} \geq \theta \wedge r_{u,i} \neq \bullet\} \tag{24}$$

$$T = \{i \in I \mid r_{u,i} \geq \theta\} \tag{25}$$

Where $L_G$ is the set of items recommended to the group $G$ in which user $u$ is, $r_{u,i}$ is the test rating of the user $u$ to the item $i$, • means that the test rating does not exist, and $\theta$ is a threshold to consider a rating as like or dislike.

Finally, we will denote *precision* and *recall* as the averaged precision and recall to each group of users, and *F1* combines the values of precision and recall.

$$precision = \frac{\sum_G precision_G}{\#G} \tag{26}$$

$$recall = \frac{\sum_G recall_G}{\#G} \tag{27}$$

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{28}$$

## C. Model Optimization of Parameters

The proposed method contains two parameters: $\alpha$, that controls the amount of overlapping of a user between user factors, and $\beta$, that fixes the amount of evidence needed to determine that an item factor is associated with a user factor. This association between an item factor and user factor allows obtaining the probability that a virtual user likes a specific item. A proper adjustment of these parameters, for each dataset, is required in order to maximize the quality.

In this experiment, we will evaluate the proposed method for different

combinations of both $\alpha$ and $\beta$ parameters. Table III contains the tested values for each parameter. To select the optimal configuration of these parameters we will: a) measure the *RMSE* and *F1*, b) recommend *15* items for each configuration of parameters, and c) select the best one. Table IV contains the remaining parameters required in this experiment.

TABLE III. TESTED MODEL PARAMETERS

| Dataset | $\alpha$ | $\beta$ |
|---|---|---|
| MovieLens | 0.1 to 1.0 step 0.1 | 1 to 10 step 1 |
| FilmTrust | 0.1 to 1.0 step 0.1 | 1 to 5 step 1 |

TABLE IV. CONFIGURATION OF THE EXPERIMENT TO OPTIMIZE THE PROPOSED METHOD PARAMETERS

| Parameter | MovieLens | FilmTrust |
|---|---|---|
| $\alpha$ | 0.4 | 0.5 |
| $\beta$ | 5 | 3 |

### D. Model Performance

To measure the performance of the proposed model we will compare it with state-of-the-art group recommendation methods. The baselines selected for this comparison are *PC* [52], [56], *RAP* [57], and *VUR* [45], [51], [55]. Some of these recommendation methods require different parameters to work. We have configured these parameters in order to maximize the quality of the recommendation in each dataset. We will evaluate both the proposed and the baselines methods using the previously defined quality measures: *RMSE* and *F1*. Table V contains the parameters in each experiment.

### E. Comparative Results and Discussion

The first set of experiments is designed to select the *most appropriate reduction dimension technique for the CF RS* field context. The tested methods are: 1) PCA, since it is the classical baseline for the machine learning reduction of dimensions field, 2) TruncatedSVD, because it is appropriate to be used in the CF sparse datasets, and 3) BNMF, since it has proved to accurately catch the CF non-linear features relations and to provide state of the art recommendation results.

To test and compare the three chosen methods we study their *cumulative explained variance* and their *user distribution*. Fig. 4a shows the cumulative explained variance results; as it can be seen, BNMF strongly outperforms TruncatedSVD and PCA. BNMF is able to catch more cumulative variance using the same number of dimensions; this means that using BNMF we expect better prediction results using

the same number of dimensions (same number of hidden factors) than using PCA or TruncatedSVD. Fig. 4b shows another relevant measure: *user distribution*; it is valuable to avoid big differences in the number of users belonging to the groups (clusters) in the CF RS. Fig. 4b shows a much better BNMF distribution of users when the number of groups is small, and somewhat better results for larger number of clusters. Overall, in the CF context, we can conclude that BNMF outperforms the tested methods and it is able to compress more information in the same number of dimensions. This is expected behavior since BNMF has been designed to outperform MF approaches in CF environments. Because of the explained results, the proposed method in this paper uses the BNMF hidden factors both to feed the clustering algorithm and to get predictions by means of the aggregated virtual users.

To improve the ***group predictions accuracy*** is a relevant objective of the paper. Fig. 5 shows the *RMSE* error results both in the MovieLens dataset (Fig. 5a) and the FilmTrust dataset (Fig. 5b). As can be seen, the proposed method RAF returns lower errors than the chosen baselines (PC, RAP, and VUR). RAF outperforms the baselines for all the tested number of groups (*x*-axis: number of clusters). As expected, the lower the number of groups the higher the number of users in each cluster; and, consequently, the higher the error. This is the behavior for both the proposed method and the most competitive baselines: RAP and PC.

It is important to note that experiments were carried out with several methods designed from the VUR one. These methods are [45], [51], and [55], which have recently been published for recommendation to groups of users. The baseline from [55] offers better results than the [45], and [51] ones. The VUR results in Fig. 5 correspond to the results obtained with the baseline [55]. Results from [45], and [51] have not been incorporated because they are worse and do not help the visualization of the methods with relevant results. The VUR method offers the worst quality results; it demonstrates that to recommend to groups of users, a method cannot be based on a just aggregation of preferences, virtual users, and similarity measures for groups of users. This is not an optimal design to recommend to large groups of users. It is logical that a virtual user that represents a very large group will tend to generate very general and not accurate predictions. This points towards our proposed method do not achieve the best results only by an aggregation of factors. The key to the success of the proposed method has been the appropriate dimensionality reduction, the aggregation of factors, and especially our probabilistic approach for groups of users that allows predicting: 1) the probability that a virtual user likes a specific item and, 2) the prediction for the virtual user (group).
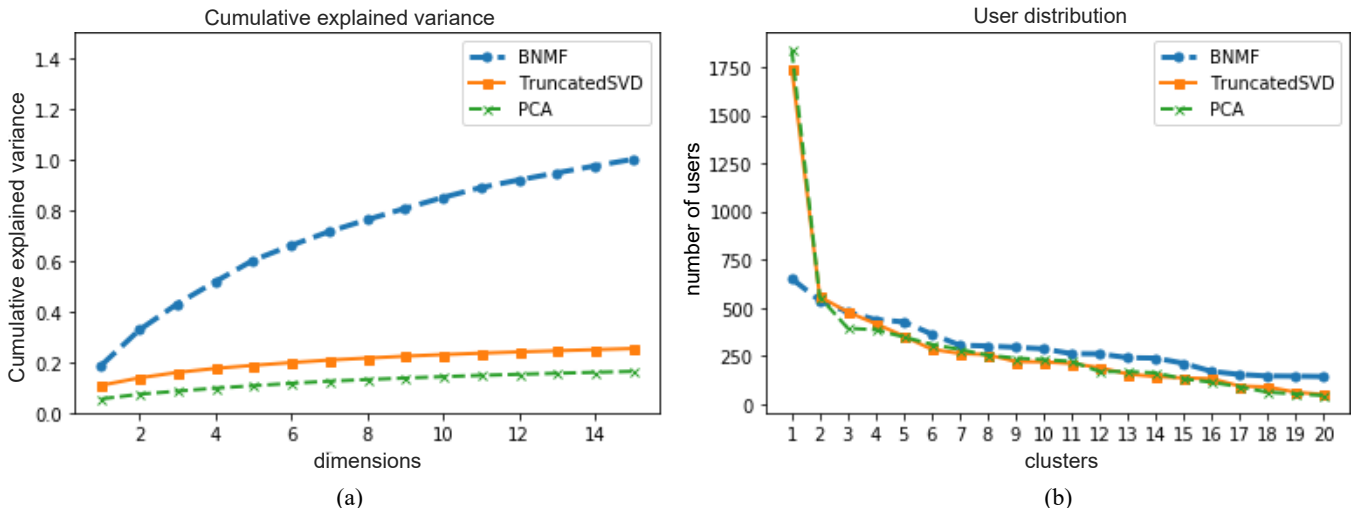


Fig. 4. Dimensionality dimensions results; a) Cumulative explained variance obtained on diverse number of dimensions; b) averaged number of users when diverse number of clusters is set.
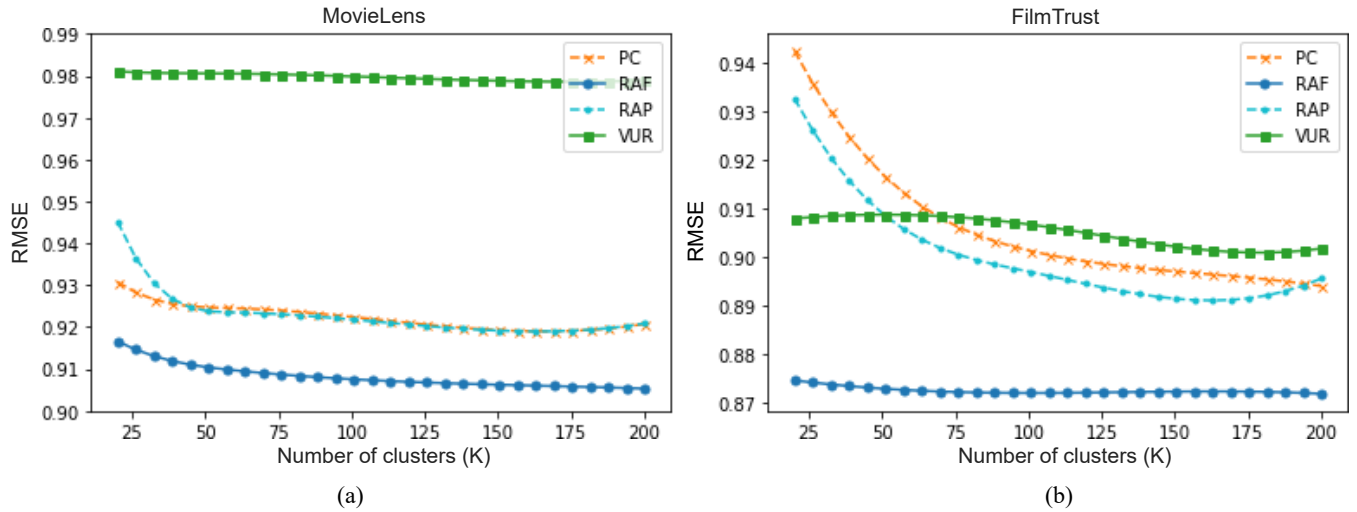
(a)  (b)

Fig. 5. Prediction accuracy results. RMSE obtained when the proposed method is run using diverse number of groups (clusters); a) MovieLens accuracy, b) FilmTrust accuracy. The lower the values, the higher the accuracy.
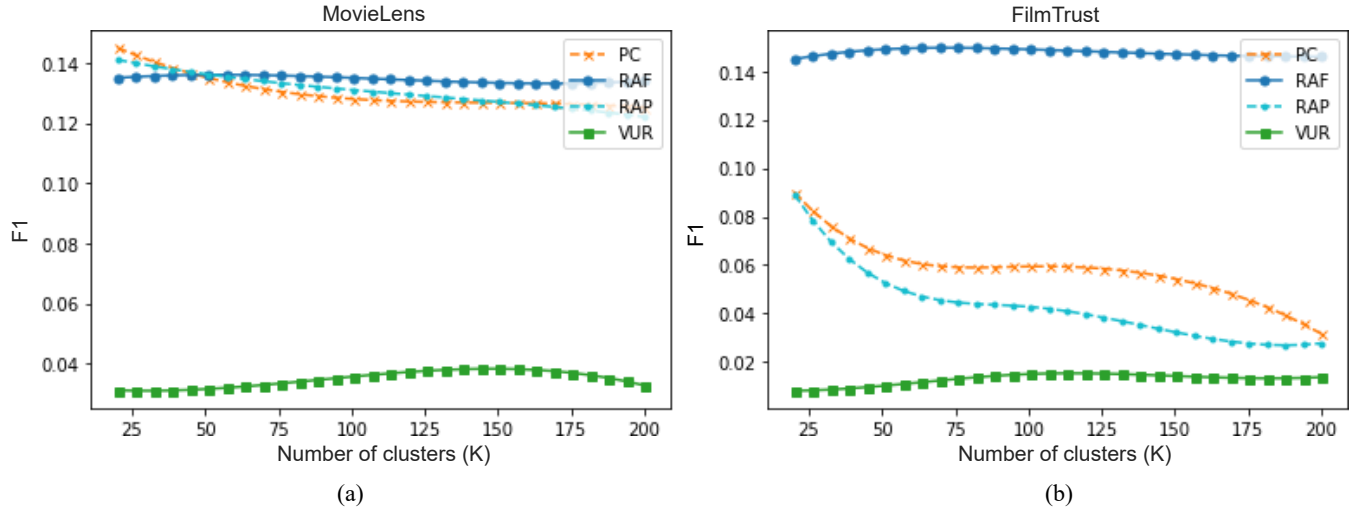


(a)  (b)

Fig. 6. Recommendation accuracy results. F1 obtained when the proposed method is run using diverse number of groups (clusters); a) MovieLens accuracy, b)

Finally, the ***recommendation accuracy*** is tested. Since prediction accuracy has been improved (Fig. 5), we also expect to improve recommendation accuracy, which is directly related to the quality of the highest predictions. To reduce the number of figures we use the F1 quality measure to join the precision and recall results. Fig. 6 is equivalent to Fig. 5, but it shows the recommendation F1 results instead of the prediction RMSE ones. In Fig. 6, higher values mean better accuracies. Overall, recommendations are improved by using the proposed method (RAF), particularly in the FilmTrust dataset, whose sparsity is much greater than the MovieLens one. The BNMF performance, when applied to the FilmTrust sparse dataset, is reflected both in the prediction and the recommendation results.

## V. Conclusions and Future Works

To make recommendations on large homogeneous and automatically detected groups of users is a challenging task that is not adequately addressed using the existing approaches to recommend to little groups of stablished users. To accurately detect collaborative filtering groups is necessary to feed the clustering process with high-level information: hidden factors obtained from ratings. This approach makes use of the abstraction level provided by the chosen dimensionality reduction method. The Bayesian non-Negative Matrix Factorization (BNMF) has

proved to be the most effective dimensionality reduction technique for this paper's objectives. Its superior cumulative deviation result shows that it provides more information, by using the same number of factors, than other representative dimensionality reduction methods. This is a key contribution of the paper since it opens the possibility to design alternative group recommendation methods based on the representative BNMF hidden factors.

The group recommendation model-based approach provided in this paper aggregates the hidden factors of each group of users to make a virtual user that represents the set of users of the group. This strategy provides two main advantages: 1) It makes the group model from the higher semantic representation of users: their factors and 2) It allows to simplify the next stages: prediction and recommendation since they can be done as if they were individual recommendations (made to the virtual users).

The proposed approach outperforms the state-of-the-art baselines used to recommend to groups of users. Prediction and recommendation results are particularly improved using the proposed method when it is applied to very sparse datasets. This is because BNMF has been designed to work on collaborative filtering sparse environments, and it provides suitable probabilistic hidden factors to feed the proposed clustering stage.

The proposed method and their results open the door to different future works, such as: a) The use of weighted aggregation approaches to obtain the virtual users, or b) The prediction stage improvement by replacing the linear dot product of hidden factors by a neural network architecture that learns the complex non-linear relations that exist between hidden factors.

## REFERENCES

[1]  J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowledge-Based Syst.*, vol. 46, pp. 109–132, 2013.

[2]  J. Bobadilla, F. Serradilla, and A. Hernando, "Collaborative filtering adapted to recommender systems of e-learning," *Knowledge-Based Syst.*, vol. 22, no. 4, pp. 261–265, 2009.

[3]  C. Yang, L. Bai, C. Zhang, Q. Yuan, and J. Han, "Bridging Collaborative Filtering and Semi-Supervised Learning: A Neural Approach for POI Recommendation," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, pp. 1245–1254.

[4]  E. Thomas, A. G. Ferrer, B. Lardeux, M. Boudia, C. Haas-Frangii, and R. A. Agost, "Cascaded Machine Learning Model for Efficient Hotel Recommendations from Air Travel Bookings," 2019.

[5]  R. Hurtado, J. Bobadilla, R. Bojorque, F. Ortega, and X. Li, "A New Recommendation Approach Based on Probabilistic Soft Clustering Methods: A Scientific Documentation Case Study," *IEEE Access*, vol. 7, pp. 7522–7534, 2019.

[6]  W.-T. Chu and Y.-L. Tsai, "A hybrid recommendation system considering visual information for predicting favorite restaurants," *World Wide Web*, vol. 20, no. 6, pp. 1313–1331, Nov. 2017.

[7]  X. Wang, X. He, L. Nie, and T.-S. Chua, "Item Silk Road: Recommending Items from Information Domains to Social Users," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2017, pp. 185–194.

[8]  X. Deng and F. Huangfu, "Collaborative Variational Deep Learning for Healthcare Recommendation," *IEEE Access*, vol. 7, pp. 55679–55688, 2019.

[9]  F. Z. Benkaddour, N. Taghezout, F. Z. Kaddour-Ahmed, and I.-A. Hammadi, "An Adapted Approach for User Profiling in a Recommendation System: Application to Industrial Diagnosis," *Int. J. Interact. Multimed. Artif. Intell.*, vol. 5, no. 3, pp. 118–130, 2018.

[10] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Adv. Artif. Intell.*, vol. 2009, 2009.

[11] J. Bobadilla, F. Ortega, A. Hernando, and Á. Arroyo, "A balanced memory-based collaborative filtering similarity measure," *Int. J. Intell. Syst.*, vol. 27, no. 10, pp. 939–946, 2012.

[12] B. Zhu, R. Hurtado, J. Bobadilla, and F. Ortega, "An efficient recommender system method based on the numerical relevances and the non-numerical structures of the ratings," *IEEE Access*, 2018.

[13] J. Bobadilla, R. Bojorque, A. H. Esteban, and R. Hurtado, "Recommender Systems Clustering Using Bayesian Non Negative Matrix Factorization," *IEEE Access*, vol. 6, pp. 3549–3564, 2018.

[14] X. Guan, C. T. Li, and Y. Guan, "Matrix Factorization with Rating Completion: An Enhanced SVD Model for Collaborative Filtering Recommender Systems," *IEEE Access*, vol. 5, pp. 27668–27678, 2017.

[15] R. Mehta and K. Rana, "A review on matrix factorization techniques in recommender systems," in *2017 2nd International Conference on Communication Systems, Computing and IT Applications (CSCITA)*, 2017, pp. 269–274.

[16] Z. Chen, L. Li, H. Peng, Y. Liu, H. Zhu, and Y. Yang, "Sparse General Non-Negative Matrix Factorization Based on Left Semi-Tensor Product," *IEEE Access*, vol. 7, pp. 81599–81611, 2019.

[17] K. Li, X. Zhou, F. Lin, W. Zeng, and G. Alterovitz, "Deep Probabilistic Matrix Factorization Framework for Online Collaborative Filtering," *IEEE Access*, vol. 7, pp. 56117–56128, 2019.

[18] A. Hernando, J. Bobadilla, and F. Ortega, "A non negative matrix factorization for collaborative filtering recommender systems based on a Bayesian probabilistic model," *Knowledge-Based Syst.*, vol. 97, pp. 188–202, 2016.

[19] X. He, H. Zhang, M.-Y. Kan, and T.-S. Chua, "Fast Matrix Factorization for Online Recommendation with Implicit Feedback," in *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2016, pp. 549–558.

[20] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural Collaborative Filtering," in *Proceedings of the 26th International Conference on World Wide Web*, 2017, pp. 173–182.

[21] D. Liang, M. Zhan, and D. P. W. Ellis, "Content-Aware Collaborative Music Recommendation Using Pre-trained Neural Networks.," in *ISMIR*, 2015, pp. 295–301.

[22] K. Yoshii, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno, "An Efficient Hybrid Music Recommender System Using an Incrementally Trainable Probabilistic Generative Model," *IEEE Trans. Audio. Speech. Lang. Processing*, vol. 16, no. 2, pp. 435–447, Feb. 2008.

[23] C. Musto, C. Greco, A. Suglia, and G. Semeraro, "Ask Me Any Rating: A Content-based Recommender System based on Recurrent Neural Networks.," in *IIR*, 2016.

[24] T. Ebesu and Y. Fang, "Neural Citation Network for Context-Aware Citation Recommendation," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2017, pp. 1093–1096.

[25] C. C. Aggarwal and C. K. Reddy, "Data clustering," *Algorithms Appl. Boca Rat. CRC Press*, 2014.

[26] Chan Young Kim, Jae Kyu Lee, Yoon Ho Cho, and Deok Hwan Kim, "VISCORS: a visual-content recommender for the mobile Web," *IEEE Intell. Syst.*, vol. 19, no. 6, pp. 32–39, Nov. 2004.

[27] L. H. Son, "A Novel Kernel Fuzzy Clustering Algorithm for Geo-Demographic Analysis," *Inf. Sci.*, vol. 317, no. C, pp. 202–223, 2015.

[28] Qing Li and Byeong Man Kim, "Clustering approach for hybrid recommender system," in *Proceedings IEEE/WIC International Conference on Web Intelligence (WI 2003)*, 2003, pp. 33–38.

[29] D. Rafailidis and P. Daras, "The TFC Model: Tensor Factorization and Tag Clustering for Item Recommendation in Social Tagging Systems," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 43, no. 3, pp. 673–688, May 2013.

[30] S. Gong, "A collaborative filtering recommendation algorithm based on user clustering and item clustering.," *JSW*, vol. 5, no. 7, pp. 745–752, 2010.

[31] R. Pan, P. Dolog, and G. Xu, "KNN-Based Clustering for Improving Social Recommender Systems," in *Agents and Data Mining Interaction*, 2013, pp. 115–125.

[32] C.-X. Zhang, Z.-K. Zhang, L. Yu, C. Liu, H. Liu, and X.-Y. Yan, "Information filtering via collaborative user clustering modeling," *Phys. A Stat. Mech. its Appl.*, vol. 396, pp. 195–203, 2014.

[33] X. Wang and J. Zhang, "Using incremental clustering technique in collaborative filtering data update," in *Proceedings of the 2014 IEEE 15th International Conference on Information Reuse and Integration (IEEE IRI 2014)*, 2014, pp. 420–427.

[34] H. WU, Y. Wang, Z. WANG, X.-L. WANG, and S.-Z. DU, "Two-Phase Collaborative Filtering Algorithm Based on Co-Clustering: Two-Phase Collaborative Filtering Algorithm Based on Co-Clustering," *J. Softw.*, vol. 21, pp. 1042–1054, 2010.

[35] C. Rana and S. K. Jain, "An evolutionary clustering algorithm based on temporal features for dynamic recommender systems," *Swarm Evol. Comput.*, vol. 14, pp. 21–30, 2014.

[36] C.-L. Liao and S.-J. Lee, "A clustering based approach to improving the efficiency of collaborative filtering recommendation," *Electron. Commer. Res. Appl.*, vol. 18, pp. 1–9, 2016.

[37] Á. M. Navarro and P. Moreno-Ger, "Comparison of Clustering Algorithms for Learning Analytics with Educational Datasets," *Int. J. Interact. Multimed. Artif. Intell.*, vol. 5, no. 2, pp. 9–16, 2018.

[38] M. K. Najafabadi, M. N. Mahrin, S. Chuprat, and H. M. Sarkan, "Improving the accuracy of collaborative filtering recommendations using clustering and association rules mining on implicit data," *Comput. Human Behav.*, vol. 67, pp. 113–128, 2017.

[39] S. Zahra, M. A. Ghazanfar, A. Khalid, M. A. Azam, U. Naeem, and A. Prugel-Bennett, "Novel centroid selection approaches for KMeans-clustering based recommender systems," *Inf. Sci. (Ny).*, vol. 320, pp. 156–189, 2015.

[40] X. Zhang, L. Zong, X. Liu, and J. Luo, "Constrained Clustering With Nonnegative Matrix Factorization," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 27, no. 7, pp. 1514–1526, 2016.

[41] N. Del Buono and G. Pio, "Non-negative Matrix Tri-Factorization for co-clustering: An analysis of the block matrix," *Inf. Sci. (Ny).*, vol. 301, pp. 13–26, 2015.

[42] T. George and S. Merugu, "A scalable collaborative filtering framework based on co-clustering," in *Fifth IEEE International Conference on Data Mining (ICDM'05)*, 2005, pp. 4 pp.-.

[43] T. Li, Y. Zhang, and V. Sindhwani, "A Non-negative Matrix Tri-factorization Approach to Sentiment Classification with Lexical Prior Knowledge," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, 2009, pp. 244–252.

[44] Q. Gu and J. Zhou, "Local learning regularized nonnegative matrix factorization," in *Twenty-First International Joint Conference on Artificial Intelligence*, 2009.

[45] F. Ortega, R. Hurtado, J. Bobadilla, and R. Bojorque, "Recommendation to Groups of Users Using the Singularities Concept," *IEEE Access*, vol. 6, 2018.

[46] L. Boratto and S. Carta, "State-of-the-Art in Group Recommendation and New Approaches for Automatic Identification of Groups," in *Information Retrieval and Mining in Distributed Environments*, A. Soro, E. Vargiu, G. Armano, and G. Paddeu, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 1–20.

[47] Y.-D. Seo, Y.-G. Kim, E. Lee, K.-S. Seol, and D.-K. Baik, "An enhanced aggregation method considering deviations for a group recommendation," *Expert Syst. Appl.*, vol. 93, pp. 299–312, 2018.

[48] S. Feng and J. Cao, "Improving group recommendations via detecting comprehensive correlative information," *Multimed. Tools Appl.*, vol. 76, no. 1, pp. 1355–1377, 2017.

[49] L. Baltrunas, T. Makcinskas, and F. Ricci, "Group recommendations with rank aggregation and collaborative filtering," in *Proceedings of the fourth ACM conference on Recommender systems - RecSys '10*, 2010, p. 119.

[50] F. Ortega, A. Hernando, J. Bobadilla, and J. H. Kang, "Recommending items to group of users using Matrix Factorization based Collaborative Filtering," *Inf. Sci. (Ny).*, vol. 345, pp. 313–324, 2016.

[51] F. Ortega, J. Bobadilla, A. Hernando, and A. Gutiérrez, "Incorporating group recommendations to recommender systems: Alternatives and performance," *Inf. Process. Manag.*, vol. 49, no. 4, pp. 895–901, 2013.

[52] L. Boratto and S. Carta, "ART: group recommendation approaches for automatically detected groups," *Int. J. Mach. Learn. Cybern.*, vol. 6, no. 6, pp. 953–980, 2015.

[53] L. Boratto and S. Carta, "The rating prediction task in a group recommender system that automatically detects groups: architectures, algorithms, and performance evaluation," *J. Intell. Inf. Syst.*, vol. 45, no. 2, pp. 221–245, Oct. 2015.

[54] S. Berkovsky and J. Freyne, "Group-based Recipe Recommendations: Analysis of Data Aggregation Strategies," in *Proceedings of the Fourth ACM Conference on Recommender Systems*, 2010, pp. 111–118.

[55] J. Masthoff, "Group Recommender Systems: Combining Individual Models," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Boston, MA: Springer US, 2011, pp. 677–702.

[56] L. Boratto, S. Carta, and G. Fenu, "Investigating the role of the rating prediction task in granularity-based group recommender systems and big data scenarios," *Inf. Sci. (Ny).*, vol. 378, pp. 424–443, 2017.

[57] M. O'Connor, D. Cosley, J. A. Konstan, and J. Riedl, "PolyLens: A Recommender System for Groups of Users," in *ECSCW 2001: Proceedings of the Seventh European Conference on Computer Supported Cooperative Work 16--20 September 2001, Bonn, Germany*, W. Prinz, M. Jarke, Y. Rogers, K. Schmidt, and V. Wulf, Eds. Dordrecht: Springer Netherlands, 2001, pp. 199–218.

[58] C. Xiong, K. Lv, H. Wang, and C. Qi, "Personalized Group Recommendation Model Based on Argumentation Topic," in *Complex, Intelligent, and Software Intensive Systems*, 2019, pp. 206–217.

[59] D. Sacharidis, "Modeling Uncertainty in Group Recommendations," in *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*, 2019, pp. 69–74.

[60] D. Cao, X. He, L. Miao, Y. An, C. Yang, and R. Hong, "Attentive Group Recommendation," in *The 41st International ACM SIGIR Conference on Research &#38; Development in Information Retrieval*, 2018, pp. 645–654.

[61] B.-H. Li *et al.*, "GRIP: A Group Recommender Based on Interactive Preference Model," *J. Comput. Sci. Technol.*, vol. 33, no. 5, pp. 1039–1055, Sep. 2018.

[62] J. Du, L. Li, P. Gu, and Q. Xie, "A Group Recommendation Approach Based on Neural Network Collaborative Filtering," in *2019 IEEE 35th International Conference on Data Engineering Workshops (ICDEW)*, 2019, pp. 148–154.

[63] A. Roy, S. Banerjee, M. Sarkar, A. Darwish, M. Elhoseny, and A. E. Hassanien, "Exploring New Vista of intelligent collaborative filtering: A restaurant recommendation paradigm," *J. Comput. Sci.*, vol. 27, pp. 168–182, 2018.

[64] J. Park and K. Nam, "Group recommender system for store product placement," *Data Min. Knowl. Discov.*, vol. 33, no. 1, pp. 204–229, 2019.

[65] D. Herzog and W. Wörndl, "A User Study on Groups Interacting with Tourist Trip Recommender Systems in Public Spaces," in *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization*, 2019, pp. 130–138.

[66] R. Logesh, V. Subramaniyaswamy, V. Vijayakumar, and X. Li, "Efficient User Profiling Based Intelligent Travel Recommender System for Individual and Group of Users," *Mob. Networks Appl.*, vol. 24, no. 3, pp. 1018–1033, Jun. 2019.

[67] T.-C. T. Chen and M.-C. Chiu, "A classifying ubiquitous clinic recommendation approach for forming patient groups and recommending suitable clinics," *Comput. Ind. Eng.*, vol. 133, pp. 165–174, 2019.

[68] A. Zawali and I. Boukhris, "A Group Recommender System for Academic Venue Personalization," in *Intelligent Systems Design and Applications*, 2020, pp. 597–606.

[69] H. J. Jeong and M. H. Kim, "HGGC: A hybrid group recommendation model considering group cohesion," *Expert Syst. Appl.*, vol. 136, pp. 73–82, 2019.

[70] S. Feng, H. Zhang, J. Cao, and Y. Yao, "Merging user social network into the random walk model for better group recommendation," *Appl. Intell.*, vol. 49, no. 6, pp. 2046–2058, Jun. 2019.

[71] T. N. T. Tran, M. Atas, A. Felfernig, V. M. Le, R. Samer, and M. Stettinger, "Towards Social Choice-based Explanations in Group Recommender Systems," in *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization*, 2019, pp. 13–21.

[72] W. Wang, G. Zhang, and J. Lu, "Hierarchy Visualization for Group Recommender Systems," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 49, no. 6, pp. 1152–1163, 2019.

[73] C.-H. Lai and Y.-C. Chang, "Document recommendation based on the analysis of group trust and user weightings," *J. Inf. Sci.*, vol. 0, no. 0, p. 0165551518819973.

[74] F. M. Harper and J. A. Konstan, "The MovieLens Datasets: History and Context," *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 4, pp. 19:1--19:19, 2015.

[75] J. Golbeck and J. Hendler, "FilmTrust: movie recommendations using trust in web-based social networks," in *CCNC 2006. 2006 3rd IEEE Consumer Communications and Networking Conference, 2006.*, 2006, vol. 1, pp. 282–286.

[76] J. Bobadilla, F. Ortega, A. Gutiérrez and S. Alonso, "Classification-based deep neural network architecture for collaborative filtering recommender systems", *International Journal of Interactive Multimedia and Artificial Intelligence*, vol. 6, no. 1, pp. 68-77, 2020.

### Remigio Hurtado

Remigio Hurtado was born in 1989. He received the B.S degree in Systems Engineering from Universidad Politécnica Salesiana, Ecuador in 2012, Master degree in Information and Software Technology, Instituto Tecnológico y de Estudios Superiores de Monterrey, México, 2014, and Master degree in Computer Science and Technology, Universidad Politécnica de Madrid, Spain, 2017. He is lecturer at the Universidad Politécnica Salesiana, Ecuador. His research interests include the recommender systems and natural language processing. He is currently member of research team in Artificial Intelligence and Assistive Technology, his team is collaborating with Universidad Politécnica de Madrid.

### Jesús Bobadilla

Jesús Bobadilla received the B.S. and the Ph.D. degrees in computer science from the Universidad Politécnica de Madrid and the Universidad Carlos III. Currently, he is a lecturer with the Department of Applied Intelligent Systems, Universidad Politécnica de Madrid. He is a habitual author of programming languages books working with McGraw-Hill, Ra-Ma and Alfa Omega publishers. His research interests include information retrieval, recommender systems and speech processing. He is in charge of the FilmAffinity.com research team working on the collaborative filtering kernel of the web site. He has been a researcher into the International Computer Science Institute at Berkeley University and into the Sheffield University. Head of the research group.

### Abraham Gutiérrez

Abraham Gutiérrez was born in Madrid, Spain, in 1969. He received the B.S. and the Ph.D. degrees in computer science from the Universidad Politécnica de Madrid. Currently, he is a lecturer with the Department of Applied Intelligent Systems, Universidad Politécnica de Madrid. He is a habitual author of programming languages books working with McGraw-Hill, Ra-Ma and Alfa Omega publishers. His research interests include P-Systems, social networks and recommender systems. He is in charge of this group innovation issues, including the commercial projects.

### Santiago Alonso

Santiago Alonso received his B.S. degree in software engineering from Universidad Autónoma de Madrid and his Ph.D. degree in computer science and artificial intelligence from Universidad Politécnica de Madrid, in 2015, where he is currently an Associate Professor, participating in master and degree subjects and doing work related with advanced databases. His main research interests include natural computing (P-Systems), and did some work on genetic algorithms. His current interests include machine learning, data analysis and artificial intelligence.