IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received July 4, 2019, accepted July 23, 2019, date of publication August 5, 2019, date of current version August 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2933048

A Collaborative Filtering Approach Based on Naïve Bayes Classifier

PRISCILA VALDIVIEZO-DIAZ[1,2], FERNANDO ORTEGA[2], EDUARDO COBOS[3], AND RAÚL LARA-CABRERA[2]

[1]Computer Science and Electronic Department, Universidad Técnica Particular de Loja, Loja 1101608, Ecuador
[2]Departamento de Lenguajes y Sistemas Informáticos, ETSI de Sistemas Informáticos, Universidad Politécnica de Madrid, Madrid, Spain
[3]Ingenio Labs, 28001 Madrid, Spain

Corresponding author: Priscila Valdiviezo-Diaz (pmvaldiviezo@utpl.edu.ec)

This work was supported by the Universidad Técnica Particular de Loja.

ABSTRACT Recommender system is an information filtering tool used to alleviate information overload for users on the web. Collaborative filtering recommends items to users based on their historical rating information. There are two approaches: memory-based, which usually provides inaccurate but explainable recommendations; and model-based, whose recommendations are more precise but hard to understand. Here we propose a Bayesian model that not only provides us with recommendations as good as matrix factorization models, but these predictions can also be explained. The model is based on both user-based and item-based collaborative filtering approaches, which recommends items by using similar users' and items' information, respectively. Experiments carried out using four datasets present good results compared to several state-of-the-art baselines, achieving the best performance using the Normalized Discounted Cumulative Gain (nDCG) quality measure and also improving the prediction's accuracy in some datasets.

INDEX TERMS Recommender systems, collaborative filtering, Naïve Bayes classifier, hybrid CF, reliability measure.

I. INTRODUCTION

Recommender Systems (RS) are becoming an alternative to facing the Information Overload problem in the web [1]. RS filter data according to an analysis of past preferences of the users. For this process, some techniques may be used, the most well-known are Content-Based Filtering (CBF), Collaborative Filtering (CF) and Hybrid Filtering [2]. CBF can be designed to recommend items similar to those that a pre-determined user used to like in the past [3]. CF is based on the assumption that users with similar taste in the past will have similar preferences in the future [4]. Hybrid Filtering combine several filtering algorithms in order to obtain a set of items or products that fit with the preferences of a user [3].

CF is the most popular implementation of a RS. These recommender systems are based on a rating matrix in which each user provides information about how much he or she likes or dislikes some items. CF methods act directly on the rating matrix to compute predictions and recommendations. CF can be subdivided into model-based and memory-based approaches. In memory-based approaches, the information to recommend is obtained directly from the rating matrix [5], these algorithms mainly divide this approach into two kinds: user-based CF and item-based CF. The user-based CF analyzes a group of users that share similar interests or experiences with the target user and recommends the items that the group generally prefers. The item-based CF recommends items that have a greater similarity with the list of items that a user liked in the past, so this approach uses similarities between the rating patterns of items to predict preferences [6]. On the other hand, in model-based approaches, a model is created from the data, which subsequently is used to make recommendations; the most popular implementation of model-based approach is Matrix Factorization (MF) [7] and its variants NMF [8], PMF [9], and BNMF [10]. In these methods, the ratings of users to items are modelled with a set of latent factors that represent features of the users and items. Nowadays, model-based methods are achieving better results in accuracy and performance.

In the context of RS, the main problem of matrix factorization is that the learnt latent space is not easy to interpret [11], so these models are not amenable to explaining their results [12]. The proposed model in [10] has been developed in order to alleviate this problem by applying a probabilistic

The associate editor coordinating the review of this manuscript and approving it for publication was Bo Sun.

approach for interpreting the factors of users and items. However, these factors remain extremely abstract for users. In this paper we address this problem creating a probabilistic model that the final users can interpret. Our model is based on a Bayesian model that combines user-based and item-based approaches. Thus, we define three different approaches based on the users' space, the items' space and the combination of both spaces.

The hypothesis this paper is that the proposed approach provides advantages for explaining recommendations, and the quality of the accuracy obtained will be equal to or better than CF baselines reported in this paper.

The rest of the paper is structured as follows: Section II presents the definition of the proposed method; Section III includes the design of the experiments to measure the quality of the performed recommendations; Section IV encloses the explanation of the recommendations; Section V shows the related work; and Section VI shows the conclusions and future work.

II. METHOD DEFINITION

In this section, we present the design and formulation of the proposed CF approach. First, we summarize the fundamentals of the Naïve Bayes Classifier and the meaning of its most important parameters. Second, we show the mathematical formulation of the proposed method.

A. METHOD DESIGN

The proposed approach has been designed based on the Naïve Bayes Classifier (NBC) [13]. We choose NBC because it allows understanding and justifying the predictions of the model in a simpler way. In addition, several studies have shown that results obtained with NBC are competitive with respect to other techniques.

NBC is a supervised multiclass classification algorithm based on applying Bayes' theorem with the "naive" assumption of conditional independence between every pair of variables.

In NBC, given a set of independent variables, $\mathrm{X} = \{\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_\mathrm{n}\}$, the posterior probability is built for each possible classes, $\mathrm{C} = \{\mathrm{c}_1, \mathrm{c}_2, \ldots, \mathrm{c}_\mathrm{m}\}$. In the proposed approach the independent variables will be the ratings of users to items and the possible classes will be each plausible rating value.

NBC calculates a classification score P(C|X), that is proportional to the posterior probability, from the prior probability of each class P(C) and the likelihood P(X|C) (1). Thus, the final classification is produced by the argument that maximizes the classification score (2). The breakdown of this equation can be found in [14].

$$P(C\,|X\,) \propto P(C) \prod_{i=1}^{n} P(x_i\,|C) \tag{1}$$

$$\hat{y} = \arg\max_{y} P(C\,|X) \tag{2}$$

Several NBC implementations have been proposed. They mainly differ by the assumptions they make regarding the likelihood distribution. In CF we can assume multinomially distributed data: each user rating is assigned to a fixed set of predefined rating values. For example, in MovieLens dataset, each rating can be classified from 1 to 5 stars.

The proposed method will classify (predict) the new ratings of a user (i.e., the possible rating value that a user would give to an item) based on the existing ratings of the dataset. The prior probabilities distributions would be computed based on the ratings of each user and item. This concept is represented by equations (3) and (6). Due to the independence assumption of NBC, the likelihood would be computed based on the ratings of a user or item regarding the ratings of another user or item respectively. We use the equations (4) and (7) to calculate the likelihood. To obtain the classification score, the prior and the likelihood are combined (equations (5) and (8)).

B. PROPOSED METHOD FORMULATION

The proposed method computes the probability that a user rates an item with a specific rating value knowing the previous ratings that exist in the dataset. This probability is computed through the prior and likelihood according to equation (1). We define three different approaches based on the computation of the prior and the likelihood:

- User-based approach: prior and likelihood are computed according to the items that each user has rated.
- Item-based approach: prior and likelihood are computed according to the ratings that each item has received.
- Hybrid approach: it integrates user-based and item-based approaches to complement each other and improve the accuracy of the model.

For this purpose, we will consider a RS as a set of U users that rate a set of I items. The rating value of a user u on item i is represented with $r_{u,i}$. The absence of rating is denoted by $\bullet$.

1) USER-BASED APPROACH

We define $P(r_i = y)$ as the probability that the item i be rated by any user as y:

$$P(r_i = y) = \frac{\#\{u \in U \,|r_{u,i} = y\} + \alpha}{\#\{u \in U \,|r_{u,i} \neq \bullet\} + \#R \cdot \alpha} \tag{3}$$

where α is a parameter to avoid 0 probabilities and $\#R$ denotes the number of plausible ratings.

We define $P(r_j = k|r_i = y)$ as the probability that the item j be rated as k knowing that the rating of item i is y:

$$P(r_j = k|r_i = y) = \frac{\#\{u \in U|r_{u,j} = k \wedge r_{u,i} = y\} + \alpha}{\#\{u \in U|r_{u,j} \neq \bullet \wedge r_{u,i} = y\} + \#R \cdot \alpha} \tag{4}$$

where α is a parameter to avoid 0 probabilities and $\#R$ denotes the number of plausible ratings.

Consequently, let $P(r_{u,i} = y)$ be the classification score that the user u rate the item i with the rating y according to the items rated by the user u.

$$P(r_{u,i} = y) \propto P(r_i = y) \prod_{j \in I_u} P(r_j = r_{u,j} | r_i = y) \tag{5}$$

where $I_u = \{i \in I | r_{u,i} \neq \bullet\}$ is the set of items rated by the user u.

2) ITEM-BASED APPROACH

We define $P(r_u = y)$ as the probability that the user u rate any item with y :

$$P(r_u = y) = \frac{\# \{i \in I \,| r_{u,i} = y\} + \alpha}{\#\{i \in I \,| r_{u,i} \neq \bullet\} + \#R \cdot \alpha} \tag{6}$$

where α is a parameter to avoid 0 probabilities and $\#R$ denotes the number of plausible ratings.

Let $P(r_v = k | r_u = y)$ be the probability that the user v rate as k knowing that the user u has rated as y.

$$P(r_v = k | r_u = y) = \frac{\#\{i \in I | r_{v,i} = k \wedge r_{u,i} = y\} + \alpha}{\#\{i \in I | r_{v,i} \neq \bullet \wedge r_{u,i} = y\} + \#R \cdot \alpha} \tag{7}$$

where α is a parameter to avoid 0 probabilities and #R denotes the number of plausible ratings.

Therefore, let $P(r_{u,i} = y)$ be the classification score that the item i be rated with y by user u according to the ratings received by the item i.

$$P(r_{u,i} = y) \propto P(r_u = y) \prod_{v \in U_i} P(r_v = r_{v,i} | r_u = y) \tag{8}$$

where $U_i = \{u \in U | r_{u,i} \neq \bullet\}$ is the set of users that has rated item i.

3) HYBRID APPROACH

Combining both user-based and item-based approaches, we can increase the number of evidences used to compute the classification score. The greater the number of evidences, the better the quality of the predictions.

Both approaches can be combined using a weighted product based on the amount of evidences used to compute the user-based and the item-based approach. Let $P(r_{u,i} = y)$ be the classification score that the user u rate the item i with the rating y:

$$P\left(r_{u,i} = y\right) \propto \left(P(r_u = y) \cdot \prod_{v \in U_i} P(r_v = r_{v,i} \,| r_u = y)\right)^{\frac{1}{1+\#U_i}} \cdot \left(P(r_i = y) \cdot \prod_{j \in I_u} P(r_j = r_{u,j} \,| r_i = y\right)^{\frac{1}{1+\#I_u}} \tag{9}$$

where $I_u = \{i \in I | r_{u,i} \neq \bullet\}$ is the set of items rated by the user u and $U_i = \{u \in U | r_{u,i} \neq \bullet\}$ is the set of users that have rated item i.

C. COMPUTE PREDICTIONS

Predictions can be computed by taking the rating value that maximizes the probability of being rated. We define $\hat{r}_{u,i}$ as the rating prediction of the user u to the item i.

$$\hat{r}_{u,i} = \arg\max_y P(r_{u,i} = y) \tag{10}$$

D. COMPUTE RELIABILITY

Reliability of predictions is an open research field in recommender systems area [15], [16]. Reliability can be defined as the certainty the recommender system has in the computed predictions, i.e, a recommender system needs to provide some certainty in the item it recommends in order to decrease the user doubt when selecting, buying or seeing that item. When reliability values are provided, each prediction is defined by the pair *<prediction, reliability>*,where the reliability represents the confidence of the model in a prediction. Using reliability, we can increase the accuracy of a RS by filtering the recommendations with low reliability, thus the system will recommend some items with high-reliability value. The idea of this measure is that the more reliable a prediction is, the less liable to be wrong.

The proposed model can easily provide reliability related to each prediction. Let $l_{u,i}$ be the reliability of the prediction of the user u to the item i.

$$l_{u,i} = \frac{P(r_{u,i} = \hat{r}_{u,i})}{\sum_{y \in R} P(r_{u,i} = y)} \tag{11}$$

where R is the set of plausible rating values of the RS.

This reliability measure can be applied to both a user-based and item-based approach and a hybrid approach.

E. RUNNING EXAMPLE

In this section, we present a running example in order to clarify how the proposed method works.

TABLE 1. A running example of the rating matrix.

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9
u_1	•	1	2	2	5	•	4	3	5
u_2	1	5	3	•	2	3	4	3	•
u_3	1	1	2	•	2	4	4	5	•
u_4	3	2	2	3	•	1	3	2	•
u_5	5	1	5	5	4	4	5	2	•

In Table 1 we show an example of a rating matrix with five users and nine items. Cells denoted with '•' indicate that the user has not rated that item.

In this running example, it is detailed how to compute the rating prediction of the user u_1 to the item i_1 ($\hat{r}_{u_1,i_1}$) using the three proposed approaches.

First, we need to compute the prior distributions. Table 2 shows the probability that the item i_1 be rated by any user with {1, 2, 3, 4, 5}, when applying a user-based approach, and the probability that the user u_1 rates any item with

TABLE 2. The prior probability of the item i_1 using the user-based approach and the user u_1 using the item-based approach.

	1	2	3	4	5
user-based	0.496297	0.002470	0.249383	0.002470	0.249383
item-based	0.143262	0.285106	0.143262	0.143262	0.285106

TABLE 3. Likelihood of the item i_1 based on the ratings of the user u_1.

		y				
		1	2	3	4	5
i_n	i_2	0.492682	0.200000	0.009523	0.200000	0.961904
	i_3	0.492682	0.200000	0.961904	0.200000	0.009523
	i_4	0.200000	0.200000	0.009523	0.200000	0.009523
	i_5	0.004878	0.200000	0.200000	0.200000	0.009523
	i_7	0.980487	0.200000	0.009523	0.200000	0.009523
	i_8	0.492682	0.200000	0.009523	0.200000	0.009523
	i_9	0.200000	0.200000	0.200000	0.200000	0.200000

$\{1, 2, 3, 4, 5\}$, when applying the item-based approach. Equation (12) details how to compute the probability that the item i_1 be rated with 1 according to equation (3).

$$P(r_{i_1}=1) = \frac{\#\left\{u \in U \,\middle|\, r_{u,i_1}=1\right\}+\alpha}{\#\left\{u \in U \,\middle|\, r_{u,i_1}\neq \bullet\right\}+\#R\cdot\alpha} = \frac{\#\{u_2,u_3\}+0.01}{\#\{u_2,u_3,u_4,u_5\}+0.05} = \frac{2.01}{4.05} = 0.496297 \tag{12}$$

Equation (13) details how to compute the probability that the user u_1 rates any item with 1 according to equation (6).

$$P(r_{u_1}=1) = \frac{\#\left\{i \in I \,\middle|\, r_{u_1,i}=1\right\}+\alpha}{\#\left\{i \in I \,\middle|\, r_{u_1,i}\neq \bullet\right\}+\#R\cdot\alpha} = \frac{\#\{i_2\}+0.01}{\#\{i_2,i_3,i_4,i_5,i_7,i_8,i_9\}+0.05} = \frac{1.01}{7.05} = 0.143262411 \tag{13}$$

Then, we need to compute the likelihood. In the user-based approach, according to equation (5), only the likelihood that relates the target item (i_1) with each item rated by the active user (u_1) must be computed. Table 3 shows the probability that the item i_1 be rated with $\{1, 2, 3, 4, 5\}$ with respect to the items rated by the user u_1. Equation (14) details how to compute the likelihood of rating the item i_1 with 1 if the item i_2 has been rated with 1.

$$P(r_{i_2}=1\,|\,r_{i_1}=1) = \frac{\#\left\{u \in U \,\middle|\, r_{u,i_2}=1 \wedge r_{u,i_1}=1\right\}+\alpha}{\#\left\{u \in U \,\middle|\, r_{u,i_2}\neq \bullet \wedge r_{u,i_1}=1\right\}+\#R\cdot\alpha} = \frac{\#\{u_3\}+0.01}{\#\{u_2,u_3\}+0.05} = \frac{1.01}{2.05} = 0.492682 \tag{14}$$

In the item-based approach, according to equation (8), only the likelihood that relates the active user (u_1) with each user that has rated the target item (i_1) must be computed.

TABLE 4. Likelihood of the user u_1 based on the ratings of the item i_1.

		v				
		1	2	3	4	5
u_j	u_2	0.00952380	0.00952380	0.00952380	0.00952380	0.00952380
	u_3	0.96190476	0.00952380	0.00952380	0.00952380	0.00952380
	u_4	0.00952380	0.49268292	0.00952380	0.96190476	0.20000000
	u_5	0.00952380	0.98048780	0.00952380	0.96190476	0.00952380

Table 4 shows the probability that the item i_1 be rated with $\{1, 2, 3, 4, 5\}$ with respect to the users that have rated that item. Equation (15) details how to compute the likelihood of rating the item i_1 with 1 if the user u_2 has rated it with 1.

$$P(r_{u_2}=1\,|\,r_{u_1}=1) = \frac{\#\left\{i \in I \,\middle|\, r_{u_2,i}=1 \wedge r_{u_1,i}=1\right\}+\alpha}{\#\left\{i \in I \,\middle|\, r_{u_2,i}\neq \bullet \wedge r_{u_1,i}=1\right\}+\#R\cdot\alpha} = \frac{\#\{\}+0.01}{\#\{i_2\}+0.05} = \frac{0.01}{1.05} = 0.0095238 \tag{15}$$

Finally, the classification score can be computed according to equations (5), (8) and (9). Table 5 contains the classification score using the user-based, item-based and hybrid approach. According to equation (10), the prediction will be 1 if user-based approach is used and 2 if item-based approach or hybrid approach is used.

TABLE 5. Classification score using the user-based, item-based, and hybrid approach. in bold the maximum classification score of each approach.

	y				
	1	2	3	4	5
user-based	**1.1355E-5**	3.1604E-8	7.8940E-11	3.1604E-8	3.7590E-12
item-based	1.19041E-7	**1.24921E-5**	1.17862E-9	1.20231E-5	4.92571E-8
Hybrid	9.93203E-3	**1.20724E-2**	8.94206E-4	1.19804E-2	1.2893E-3

To compute the reliability of these predictions, equation (11) must be applied. Using the hybrid approach, the prediction is 2, so the reliability value of this prediction we can compute as follows:

$$l_{u_1,i_1} = \frac{P(r_{u_1,i_1}=2)}{\sum_{y=1}^{5} P(r_{u_1,i_1}=y)} = \frac{1.20724\text{E-}2}{0.03616834} = 0.333784 \tag{16}$$

This value means the algorithm has a reliability of 33.4% that the prediction is correct.

F. NBCF ALGORITHM

In this section we present the algorithm used to implement the NBCF method. In order to reduce the computational

Algorithm 1 NBCF Algorithm

input : $(r_{u,i})$, (α)
output : $(pup_{u,y})$, $(pip_{i,y})$, $(cup_{v,k,u,y})$, $(cip_{j,k,i,y})$
temp : (uc_u), (ic_i), $(ijc_{j,i,y})$, $(uvc_{v,u,y})$
Initialize $pup_{u,y}$, $pip_{i,y}$, $cup_{v,k,u,y}$ *and* $cip_{j,k,i,y}$ with (α)
Initialize uc_u, ic_i, $ijc_{j,i,y}$ and $uvc_{v,u,y}$ with $\#R \cdot \alpha$

for *each user u* **do**
  for *each item i rated by user u* **do**
    $y \leftarrow r_{u,i}$
    $pup_{u,y} \leftarrow \frac{uc_u \cdot pup_{u,y}+1}{uc_u+1}$
    $uc_u \leftarrow uc_u + 1$
    $pip_{i,y} \leftarrow \frac{ic_i \cdot pip_{i,y}+1}{ic_i+1}$
    $ic_i \leftarrow ic_i + 1$
    for *each item j rated by user u* **do**
      $k \leftarrow r_{u,j}$
      $cip_{j,k,i,y} \leftarrow \frac{ijc_{j,i,y} \cdot cip_{j,k,i,y}+1}{ijc_{j,i,y}+1}$
      $ijc_{j,i,y} \leftarrow ijc_{j,i,y} + 1$
    end
    for *each user v that have rated item i* **do**
      $k \leftarrow r_{v,i}$
      $cup_{v,k,u,y} \leftarrow \frac{uvc_{v,u,y} \cdot cup_{v,k,u,y}+1}{uvc_{v,u,y}+1}$
      $uvc_{v,u,y} \leftarrow uvc_{v,u,y} + 1$
    end
  end
end

complexity of the algorithm, a memorization approach has been followed. Algorithm 1 shows the steps to compute the probabilities that are necessary to obtain the predictions, hence the algorithm builds the model that should be used when computing the predictions. This algorithm computes those probabilities in an iterative way, storing the values to avoid recalculating them later.

For the sake of simplicity, variables of the algorithm have been defined using short identifiers. The reader should take into account that $(pup_{u,y})$, $(pip_{i,y})$ are prior probabilities of users and items according to equations 3 and 6, respectively, while $(cup_{v,k,u,y})$, $(cip_{j,k,i,y})$ are the conditioned probabilities of users and items, respectively, based on equations 4 and 7. The remaining variables (uc_u), (ic_i), $(ijc_{j,i,y})$, $(uvc_{v,u,y})$ are counters of observations used to compute the aforementioned probabilities.

III. EXPERIMENTAL RESULTS

A. EXPERIMENTAL SETUP

The designed experiments compare the proposed approach with several model-based baselines, using testing quality measures and processing four public CF datasets (MovieLens, FilmTrust, Yahoo, and BookCrossing), which are commonly used in the recommender systems field. For the BookCrossing dataset, we consider only the book rating explicit information from the original dataset. The main parameters of these datasets can be shown in Table 6.

TABLE 6. Main properties of the datasets used in the experiments.

Datasets	#users	#items	#ratings	rating scale	sparsity
MovieLens	6,040	3,701	1,000,209	1-5	95,53%
FilmTrust	1,508	2,071	35,494	0.5-4	98,86%
Yahoo	7,210	4,000	167,602	1-5	99,42%
BookCrossing	77,805	185,963	432,628	1-10	99,99%

The proposed model has been designed using a probabilistic approach in order to facilitate the explanation of its results. This concept has previously been applied in the RS field. We have selected the most popular recommendations methods that use a probabilistic approach to explain their results as baselines. Experiments have made use of the Collaborative Filtering for Java framework (CF4J) [20]. In all experiments, certain hyperparameters need to be defined in accordance with the compared CF method. These are shown into each baseline. The hyperparameters values involved in the cross-validation experiments have been selected in order to maximize the accuracy of algorithms for all quality measures utilized. Testing and training sets percentages are same for all tested datasets: test users = 20%, test items = 20%, training users = 80%, training items= 80%. The baselines selected are:

- **Bayesian Non-negative Matrix Factorization (BNMF).** BNMF associates a vector of K component to each user and each item. The components of these vectors lie within [0,1] with an understandable probabilistic meaning, they allow find out some groups of users with the same tastes, as well as justify and understand the recommendations this technique provides [10]. The hyperparameters set for this method are:
  a) Number of latent factors (k): 6 for MovieLens and BookCrossing; and 8 for FilmTrust and Yahoo.
  b) Number of iterations: 150 with MovieLens; 50 for FilmTrust; and 120 for Yahoo and BookCrossing.
  c) Cluster overlapping probability (α): 0.8 for all tested datasets.
  d) Number of evidences to belong to a cluster (β): 5 for MovieLens and Yahoo; 4 for FilmTrust and BookCrossing.
- **Biclustering Hybrid Collaborative Filtering Bi-CF.** Bi-CF is a hybrid approach that combines user-based CF and item-based CF. It also uses the biclustering technique to reduce dimensionality. It adopts the user-based (UBCF) and item-based CF (IBCF) schemes based on the computed similarity respectively. Finally, it combines the resultant predictions of each model to make final predictions [17]. The hyperparameters used to perform the experiments with Bi-CF are:

a) Number of iterations: 50 for MovieLens; 20 for FilmTrust; 30 for Yahoo and BookCrossing.
b) Number of clusters: 4 for MovieLens; 2 for FilmTrust; 3 for Yahoo and BookCrossing.
c) Regularization parameters (α and γ): $\alpha = 0.001$ for MovieLens and Yahoo; $\alpha = 0.01$ for FilmTrust and BookCrossing; $\gamma = 0.8$ for all datasets.

- **Gaussian-Gamma Model (GGM).** GGM technique is based on Naive Bayesian, which explores two different methods of modelling probabilities. A Gaussian model for rating behavior, with the addition of a Gaussian-Gamma prior, this model maintains good performance even when data is sparse, and Multimodal model, which is equivalent to taking maximum likelihood estimates of the parameters of a multinomial distribution [18].
- **Improved Naïve Bayesian Method (INBM).** INBM method is based on Naive Bayesian, which has similar complexity to the original Naive Bayesian method. However, it has an adjustment of the Independence, which makes it possible to be applied to the instance where conditional independence assumption is not obeyed strictly. [19].
- **Non-Negative Matrix Factorization (NMF).**In the NMF model a ratings matrix is factorized into two matrices W and H. The values of these two matrices must always be greater than 0. The matrices W and H are found by minimizing the approximation error subject to the nonnegative constraints [8]. We have selected NMF as another baseline, because, unlike PMF which includes negative factors (which makes it difficult to justify the recommendations), NMF allows us to work only with positive factors, which facilitates to explain the recommendations. The hyperparameters required by this method are:
  a) Number of iterations: 150 for MovieLens and Yahoo; 100 with FimlTrust; 300 for BookCrossing.
  b) Number of latent factors (k): 15 for Movielens and Yahoo; 8 for FilmTrust; 12 for BookCrossing.

B. QUALITY MEASURES

In our experiments, we test predictions and recommendations accuracy. The selected quality measures for this purpose are: Mean Absolute Error (MAE), to measure the quality of the predictions; Precision, Recall and normalized Discounted Cumulative Gain (nDCG), to measure the quality of recommendations.

According to [21], *MAE* is defined as the averaged MAE_u for all the users of the *RS*. The following equation is used:

$$MAE = \frac{MAE_u}{\#U} \tag{17}$$

where MAE_u is the mean absolute difference between the test ratings of the user u and the predicted ones.

$$MAE_u = \frac{\sum_{i \in \hat{I}_u} \left| r_{u,i} - \hat{r}_{u,i} \right|}{\#\hat{I}_u} \tag{18}$$

$\hat{I}_u$ is the set of the test items rated by the user u.

precision is defined as the averaged $precision_u$ for all the users of the RS:

$$precision = \frac{precision_u}{\#U} \tag{19}$$

where $precision_u$ is the proportion of the N items recommended to the user u that are relevant to him/her. We used the following equation:

$$precision_u = \frac{\#\left\{ i \in R_u^N \left| r_{u,i} \geq \theta \right. \right\}}{N} \tag{20}$$

In this case, R_u^N is the set of N items recommended to the user u and θ is a threshold value indicating if a recommendation is relevant or not.

recall is defined as the averaged $recall_u$ for all the users of the RS:

$$recall = \frac{recall_u}{\#U} \tag{21}$$

where $recall_u$ is the proportion of relevant items recommended to the user u with respect to the total relevant items rated:

$$recall_u = \frac{\#\left\{ i \in R_u^N \left| r_{u,i} \geq \theta \right. \right\}}{\#\left\{ i \in \hat{I}_u \left| r_{u,i} \geq \theta \right. \right\}} \tag{22}$$

nDCG is defined as the averaged $nDCG_u$ for all the users of the RS:

$$nDCG = \frac{nDCG_u}{\#U} \tag{23}$$

where $nDCG_u$ is the normalized relevance of N recommendations provided to user u based on its position in the recommendation list:

$$nDCG_u = \frac{DCG_u}{IDCG_u} \tag{24}$$

$IDCG_u$ is theDCG_u of ideal ranking order, that is, the actual rankings of the user in the test set.

$$DCG_u = \sum_{p=1}^{N} \frac{2^{r_{u,x_{u,p}}} - 1}{\log_2(p+1)} \tag{25}$$

$$IDCG_u = \sum_{p=1}^{\#\hat{I}_u} \frac{2^{r_{u,y_{u,p}}} - 1}{\log_2(p+1)} \tag{26}$$

where $x_{u,p}$ stand for the item recommended at the p-th position if items recommended to user u are sorted from higher to lower prediction ($\hat{r}_{u,i}$) and $y_{\mathrm{u,p}}$ stand for the item at the p-th position if test items rated by user u ($\hat{I}_u$) are sorted by its rating value ($r_{\mathrm{u,i}}$).

C. CLASSIFIER PERFORMANCE

The proposed method is a multiclass classification algorithm. It classifies each pair *<user, item>* into a rating value. To analyze the quality of this classification we have computed the confusion matrix of user-based, item-based and hybrid approaches. Fig. 1 contains the classifier performance for

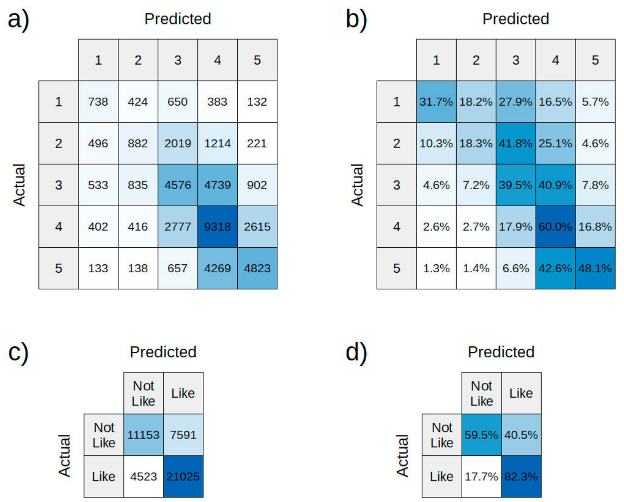

FIGURE 1. (a) Confusion matrix; (b) normalized confusion matrix; (c) confusion matrix discretizing the rating (not like = {1, 2, 3}, like = {4, 5}); (d) normalized confusion matrix discretizing the rating. MovieLens-1M dataset.

TABLE 7. Mean absolute error (mae) of the predictions performed to the test ratings.

Method	MovieLens	FilmTrust	Yahoo	BookCrossing
NBCF (user)	0,78369	0,77963	0,68690	1,80944
NBCF (item)	0,79162	0,85733	1,08200	1,61323
NBCF (hybrid)	0,70915	0,74948	**0,66715**	**1,48323**
BNMF	**0,70724**	**0,66523**	1,04870	1,91935
GGM	1,22667	1,06041	0,82387	2,31250
INBM	0,89660	1,00780	1,6406	1,97263
Bi-CF	1,14589	1,50488	1,85349	1,76251
NMF	0,77006	0,78844	1,20069	1.791998

NBCF (hybrid) in MovieLens. We adopt a confusion matrix normalization technique because it provides a better interpretation of the data.

From fig 1. (a) We can observe that there are a higher number of predictions with high ratings (4 and 5), which means that the users usually rate the items in which they are most interested. In fig. 1 (b), for a low percentage of the observations, NBCF model incorrectly predicted a rating of either 1 or 2 or 3 when the actual rating is 4 or 5, this means that users are not recommended items that are of interest to them. However, for a high percentage of the observations, the NBCF model correctly predicted a rating of 4 or 5 when the actual rating is the same. In fig 1 (d) we can observe that the 59% of items have been correctly identified as not Like and the 82,3% have been correctly identified as Like.

D. RECOMMENDER SYSTEM PERFORMANCE

In this section, we include the empirical results obtained from the comparison of the proposed model with the baselines. The hyperparameter α of the proposed approach (NBCF) has been fixed to 0.01 for all the experiments.

Table 7 contains the MAE for each dataset. It is observed that NBCF (hybrid) accuracy is better than all the CF baselines in two of the datasets (Yahoo and BookCrossing). In reference to MovieLens the result of MAE is close to that obtained with BNMF. On the other hand, in FilmTrust, NBCF is the second model with the best result in MAE obtained.

Therefore, the proposed method NBCF (hybrid) provides a significant improvement in MAE with respect to the most CF baselines we use. We believe that the reason why NBCF does not obtain the best results in FilmTrust dataset is because the data volume of FilmTrust is relatively small, resulting in poor performance on parameter estimation.

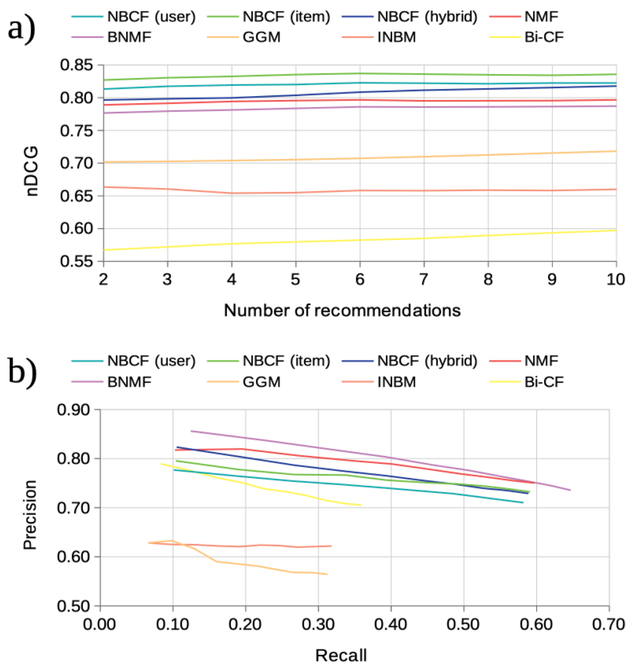

FIGURE 2. (a) Normalized Discounted Cumulative Gain (nDCG), (b) precision & recall of each recommendation method for MovieLens-1M dataset.

Fig. 2 contains the nDCG and the Precision & Recall values for MovieLens dataset. Precision and Recall have been tested using the relevance threshold at value 4 ($\theta = 4$) to discriminate if a recommendation is relevant or not. In fig. 2(a), we can observe that results of nDCG show that proposed approach NBCF (item), NBCF (user,) NBCF (Hybrid) outperforms baseline approaches in terms of ranking accuracy. We can find that the NMF and BNMF approaches perform worse than NBCF but better than INBM, another Bayesian method.

Fig. 2 (b) shows that BNMF provides better Precision & Recall than tested CF approaches. NBCF (Hybrid) achieves recommendation accuracy greater than the considered baselines (except for the BNMF and NMF methods). Additionally, NBCF (item) and NBCF (user) present better recommendation accuracy than tested baselines (GGM, INBM, Bi-CF).

Besides, analyzing the performance of the three proposed approaches, we can conclude from fig. 2 that NBCF (hybrid) approach is better in most of the metrics calculated, for example: in MAE, precision and recall. However, nDCG results show that NBCF (items) is better than the other two proposed approaches.

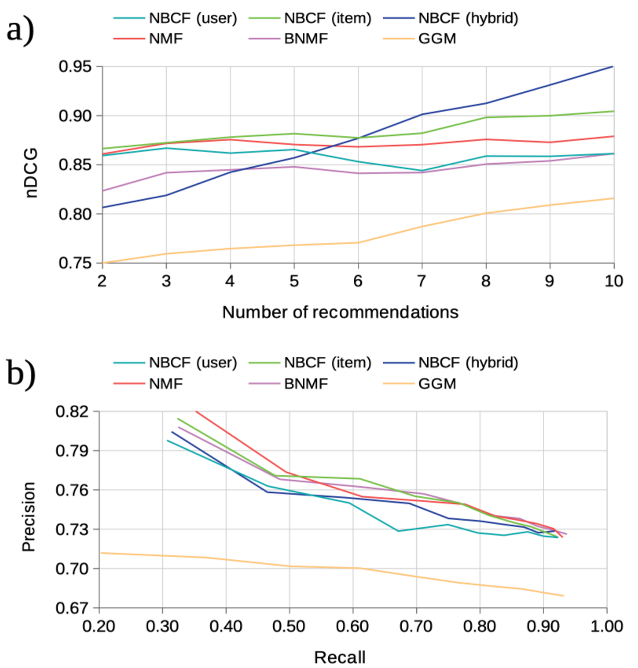

FIGURE 3. **(a) Normalized Discounted Cumulative Gain (nDCG), (b) precision & recall of each recommendation method for FilmTrust dataset.**

Fig. 3 contains the nDCG and the Precision & Recall values for FilmTrust dataset. Precision & Recall have been tested using the relevance threshold at value 3 to discriminate if the test rating is relevant or not ($\theta = 3$). Examining Fig. 3 (a), we can observe that proposed approaches provide higher nDCG than the baselines we used. NBCF (user) presents similar nDCG than BNMF method when the number of recommendations increases, and it has similar behavior than NMF method when the number of recommendations decreases. INBM and Bi-CF show an nDCG worse than the other state-of-the-art CF methods.

From Fig. 3 (b), we can observe that NBCF, NMF and BNMF provide better Precision & Recall than others tested CF approaches. Analyzing this figure, we observe a Precision & Recall behavior equivalent to that obtained in nDCG, where the behavior of our proposed approach NBCF shows better results than CF baselines. The INBM and Bi-CF methods have been removed from this figure in order to compact the results display area. Their values of nDCG and Precision were always lower than the minimum value defined in the y-axe of both graphs.

Additionally, we compared the three proposed approaches, getting the following outcomes. The MAE of NBCF (hybrid) achieves better results than the other two proposed approaches, whereas the precision and recall is better with NBCF (items) and NBCF (users). On the other hand, when the number of recommendations increases, nDCG is better with NBCF (hybrid) approach.

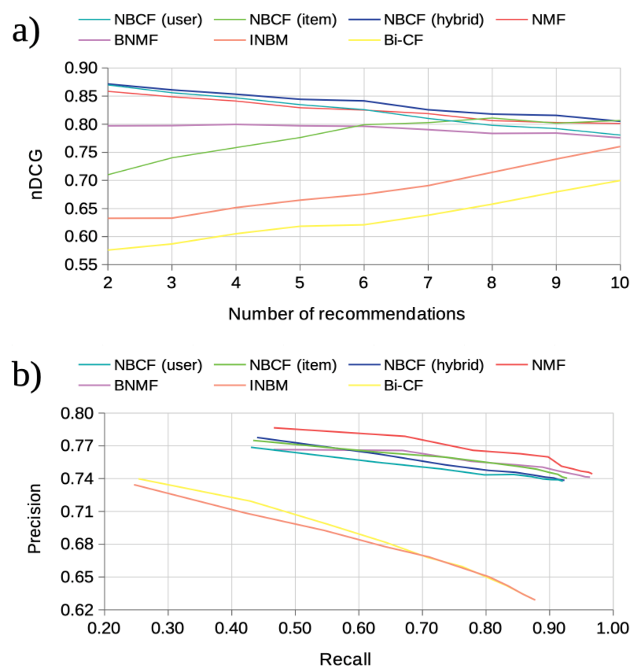

FIGURE 4. **(a) Normalized Discounted Cumulative Gain (nDCG), (b) precision & recall of each recommendation method for Yahoo dataset.**

Fig. 4 contains the nDCG and the Precision & Recall values for Yahoo dataset. The relevance value is $\theta = 4$. We can observe that NBCF (user) and NBCF (hybrid) show a better performance in nDCG that selected baselines. GGM performs worse in Yahoo dataset.

Fig. 4 (a) shows also that when the number of recommendations increases, NBCF (items) and NMF present similar performance in nDCG than others proposed methods. We can observe that NBCF, as well as NMF, provide better Precision & Recall than any other tested CF approach. Similarly, as seen in Fig. 4, nDCG is better in NBCF (Hybrid) compared to NBCF (items) and NBCF (user). Also, the precision and recall of the three proposed approaches present a result almost similar among them. We can see the superiority achieved in MAE of NBCF (hybrid) with respect to the other proposed approaches. The GGM method has been removed from this figure in order to compact the results display area. It values of nDCG and Precision were always lower than the minimum value defined in the y-axe of both graphs.

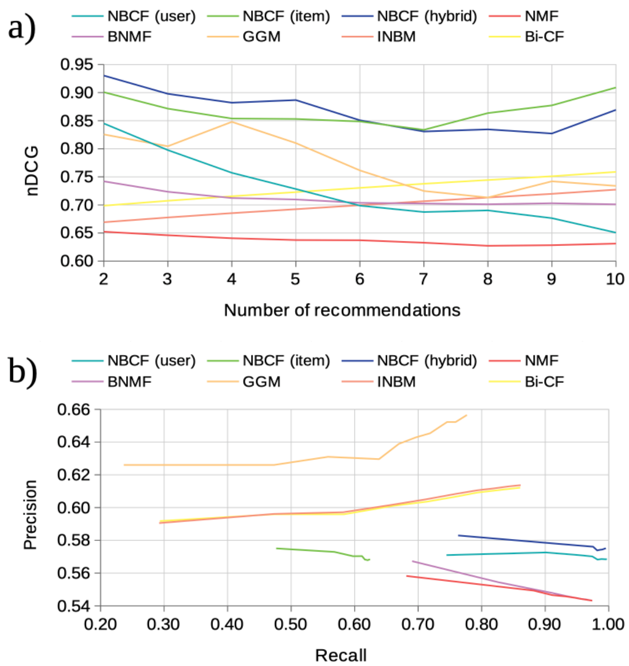

FIGURE 5. (a) Normalized Discounted Cumulative Gain (nDCG), (b) precision & recall of each recommendation method for BookCrossing dataset.

Fig. 5 contains the nDCG and the Precision & Recall values for BookCrossing dataset. Precision and Recall have been tested using the relevance threshold at value 8 ($\theta = 8$). In this dataset, we can observe that NBCF (hybrid) and NBCF (item) provide better results for nDCG compared to CF baselines.

NMF presents a worse result than the other methods. Fig. 5 (b) shows that, unlike other datasets, in BookCrossing the Precision & Recall is better for GGM, INBM and Bi-CF methods. Moreover, the proposed approaches show an improvement in Precision and Recall with respect to the BNMF and NMF methods.

Additionally, comparing the three proposed approaches, in fig. 5 (a), we can observe that when the number of recommendations is low, NBCF (hybrid) is better than the other proposed approaches, whereas when the number of recommendations is high NBCF (items) is better. As in most dataset, NBCF (hybrid) achieve a particular improvement in precision and recall compared to NBCF (items) and NBCF (user) approaches.

Experimental results show an improvement in nDCG using NBCF for all datasets present in Table 6. In summary, analyzing Fig. 2 to 5 we can establish the superiority of the proposed approaches over baselines we used: These provide the best tradeoff between nDCG and Precision & Recall measures.

Additionally, based on the results, it seems that NBCF (hybrid) is better than the NBCF (items) and NBCF (user) in terms of MAE, and it is better in nDCG for FilmTrust when the number of recommendations increases, with Yahoo dataset NBCF (hybrid) shows a minimal improvement than the other proposed methods. For BookCrossing NBCF (hybrid) is better in nDCG when the number of recommendation decrease. In terms of precision and recall, NBCF (hybrid) achieves better results in two of the dataset we used to compare to the other two proposed approaches.

E. RELIABILITY MEASURE

The proposed method provides a reliability value that represents the confidence of the model in the predictions that it performs. Fig. 6 contains the percentage of perfect predictions regarded to a fixed reliability value in MovieLens dataset for NBCF (hybrid). We can see that, as reliability increases, the percentage of perfect predictions increases (i.e. the percentage of predictions with the error equal to zero). When the reliability is over 0.35, 50% of the predictions are perfect. When the reliability is over 0.55, 70% of the predictions are perfect. This experiment demonstrates that the reliability value provides additional information about the predictions. Additionally, it provides users with valuable information about the reliability of the recommendations. We expect that if the recommender system recommends an item to a user with a high prediction, this item will satisfy the user.

F. COMPUTATIONAL COMPLEXITY ANALYSIS

The computational complexity of the algorithm 1 is $O(U \cdot \bar{I} \cdot (\bar{I} + \bar{U}))$, with $\bar{U}$ and $\bar{I}$ representing the average number of users that rated an item and the average number of items that were rated by a user, respectively. Note that $\bar{I} \ll I$ and $\bar{U} \ll$ U in the context of CF problems.

NBCF has a similar computational complexity than MF algorithms. NMF [8] and BNMF [10] have a computational complexity of $O(N \cdot K \cdot U \cdot \bar{I})$. Both computational complexities share the terms U and $\bar{I}$, so the differences lie in the comparison of $N \cdot K$ with respect to $\bar{U}$ and $\bar{I}$. For example, in MovieLens dataset, N = 150, K $\approx$ 10, $\bar{U} \approx 260$, and $\bar{I} \approx 160$. In this case, NBCF would have a complexity slightly lower than MF. In other datasets, we can find that NBCF would have a complexity slightly higher than MF. We can conclude that the computational complexity of NBCF and MF are approximately the same.

IV. RECOMMENDATIONS EXPLANATION

The proposed method allows for explaining the predictions. For any prediction, it is enough to see which other ratings have influenced more in this prediction.

To explain a recommendation to the user u (we assume that recommendations will like him), we consider the case in which the system has recommended the item i with an estimated rating of $\hat{r}_{u,i}$.

If the approach is "user-based":

    Sort all items rated by user u according to their likelihood within the item i from highest to lowest
    Repeat until you have P evidences:
        Pop the first item j from the list
        If the user u has rated positively the item j:
            Add item j to the evidences

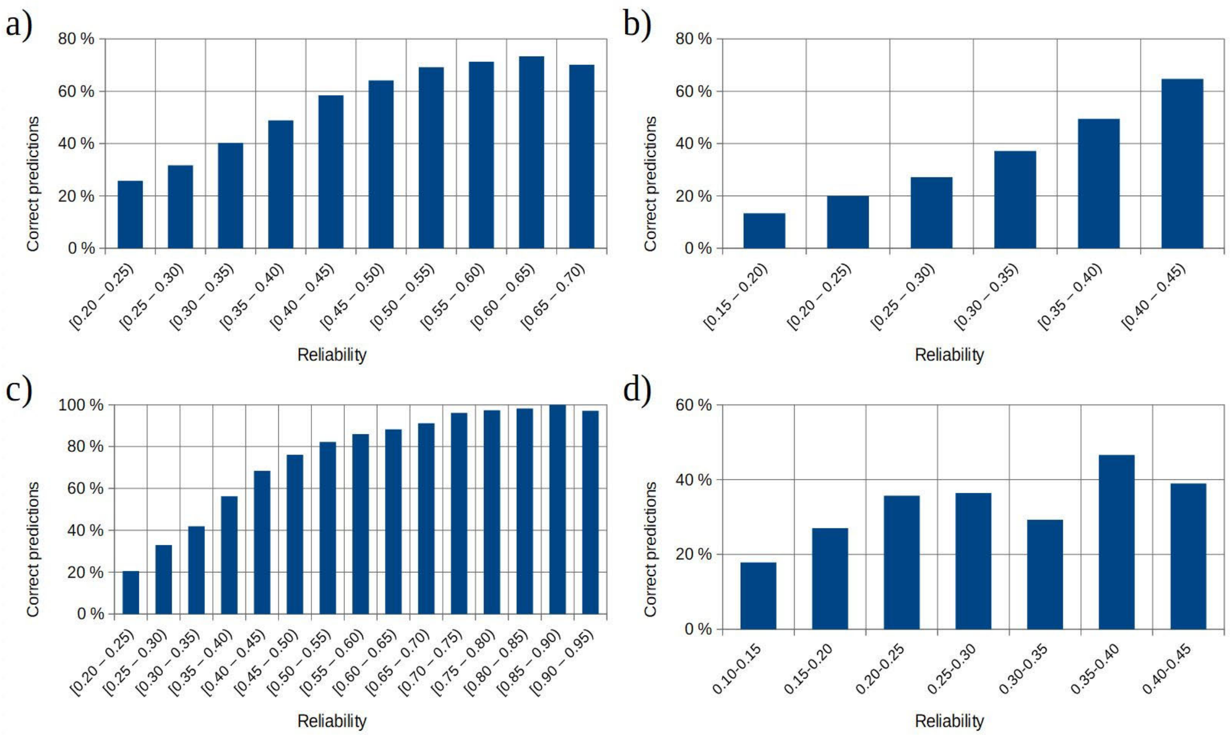

FIGURE 6. **Reliability of the recommendation in (a) movielens dataset, (b) filmtrust dataset, (c) yahoo dataset, and (d) bookcroosing dataset.**

Justify your recommendation with the next sentence: *"you will like the item i because you liked the items <complete with the P evidences>".*

If the approach is "item-based":

Sort all users that have rated the item *i* according to their likelihood within the user *u* from highest to lowest

Repeat until you have Q evidences:

Pop the first user *v* from the list

If the user *v* has rated positively the item *i*:

Add user *v* to the evidences

Justify your recommendation with the next sentence: *"you will like the item i because it liked to the users <complete with the Q evidences> who share interests with you".*

If the approach is hybrid

Both previous approaches are combined by adjusting the P and Q values according to the needs

Justify your recommendation with the next sentence: *"you will like the item i because you liked the items <complete with the P evidences> and it liked to the users <complete with the Q evidences> who share interests with you".*

V. RELATED WORK

Probabilistic models have been widely studied within CF because they are able to explain the recommendations made by these recommender systems. The most significant works related to this subject are: A probabilistic model [22] which integrates the items, the users, as well as the associations between them into a generative process. They derive a progressive algorithm to construct an ensemble of collaborative filters. Likewise, [23] presents a probabilistic item embedding model, that learns item representations from click data, and a model named EMB-MF, that combines the probabilistic item embedding and PMF for coupling the item representations of the two models.

In [24], their authors propose a local probabilistic matrix factorization (LPMF) algorithm, which divides the entire matrix into a certain number of local matrices and combines these local optimal solutions in a weighted manner. It learns local models based on specific local matrices.

BNMF [10] is a technique used for predicting the tastes of users in RS based on CF. It is based on factorizing the rating matrix into two non-negative matrices whose components have an understandable probabilistic meaning.

Reference [18] shows two different methods of modeling probabilities: a Gaussian model for rating behavior, with the addition of a Gaussian-Gamma prior, and Multimodal model, which is equivalent to taking maximum likelihood estimates of the parameters of a multinomial distribution.

The Improved Naive Bayesian Method [19] provides a new simple solution to the lack of independence other than Bayesian networks. It has an adjustment of the independence, which makes it possible to be applied to the instance where conditional independence assumption is not obeyed strictly.

Two probabilistic models are presented in [25] with prior parameters that the user can set to encourage the model to have a desired size and shape, to conform with a domain-specific definition of interpretability. They provide an approximate inference method that uses association rule mining and a randomized search algorithm to find optimal Bayesian Rule Sets maximum a posteriori model.

In [26] authors present a distributed memo-free variational inference method for large-scale matrix factorization problem, which is based on a newly derived principle of distributed variational Bayesian inference.

Reference [27] proposes a movie genre Bayesian prediction model based on user ratings. They treat ratings as a feature vector, apply a multivariate Bernoulli model to estimate of a movie being assigned a certain genre, and evaluate the classification score of the genre of a given movie by using the Bayes rule.

A natural inference model based on uncertainty rules to offer to non-registered users the possibility of inferring themselves their own recommendations is proposed in [28]. This is mathematically formalized by means of a probabilistic model that simulates the forward reasoning based on rules.

Reference [29] presents two generative processes of ratings are formulated by probabilistic graphical models with corresponding latent factors, the partial latent factor model (PLFM) and biased latent factor model (BLFM). The full Bayesian frameworks of such graphical models are proposed as well as the variational inference approaches for the parameter estimation. Reference [30] uses the Bayesian approach into the MF model; the authors propose two novel latent factor models, which incorporate both socially-influenced feature value discrepancies and socially-influenced conditional feature value discrepancies.

To fusion the user's space with that of the items usually provide more accurate recommendations than independent approaches. Some works that combine both approaches have been proposed in the literature, more recently: A hybrid approach that combines user-based CF (UBCF) and item-based CF (IBCF) is presented in [17]. It uses the biclustering technique to reduce dimensionality. The performance of the fusion of these approaches is called Bi-CF, which combines the resultant predictions of each model to make final predictions. Likewise [31] describes the fusion of user-based (UbCF) and item-based (IbCF) CF to minimize error in prediction. Predictions from UbCF and IbCF are combined through simple and weighted averaging and performance of these fusion approaches is compared with the performance of UbCf and IbCF when implemented individually. In the same way, [32] presents the IU-PMF model fusing Item Similarity and User Similarity into the PMF model to make more personalized and accurate recommendations. It combines the merits of both methods.

A hybrid recommendation approach and a framework using Gaussian mixture model and matrix factorization technology is showed in [33], where the improved cosine similarity formula is used to get users' neighbors, and initial ratings on unrated items are predicted. Users' ratings on items are converted into users' preferences on items' attributes to reduce the problem of data sparsity. The obtained user-item-attribute preference data is trained through the Gaussian mixture model to classify users with the same interests into the same group.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we have presented the design of a Bayesian model that provides recommendations as good as the matrix factorization models and that allows us to justify the recommendations. We have combined approaches based on the space of users and items within a single model. In addition, our model made possibly get reliability values associated with the predictions.

We have tested the proposed approach using four public datasets and a large set of model-based CF baselines. The results of the experiments indicate the proposed approach gets better precision results, especially when the number of recommendations is low.

The proposed approach offers better characteristics than the tested baseline methods. Due, proposed approach provides improvements in its results, these can be suitable for making comparisons between different methods and various datasets. The results also show the superiority of the proposed approach with respect to the baselines in nDCG. Likewise, it presents significant improvements in the prediction's accuracy in two of the tested datasets.

Consequently, the results obtained with NBCF are considered acceptable, since the model allows estimate the rating an item could have.

The proposed method could be extendable through the integration of attributes information of items or users, contextual information, etc. This would be achieved by modifying the computations of the conditioned probability.

The following topics are proposed as future work: (a) To apply other measures (novelty, diversity, serendipity, etc.) to test the proposed approach; (b) To extend the proposed method to groups of users; (c) To study the parallelization of the algorithm to improve its scalability; and (d) To extend the proposed method by integrating social network information, attributes information of items, and various contextual information.

REFERENCES

[1] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, "Recommender system application developments: A survey," *Decis. Support Syst.*, vol. 74, pp. 12–32, Jun. 2015.

[2] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowl. Syst.*, vol. 46, pp. 109–132, Jul. 2013.

[3] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds., *Recommender Systems Handbook*. Boston, MA, USA: Springer, 2011.

[4] N. P. Kumar and Z. Fan, "Hybrid user-item based collaborative filtering," *Procedia Comput. Sci.*, vol. 60, no. 1, pp. 1453–1461, 2015.

[5] 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*, vol. 6, pp. 49935–49954, 2018.

[6] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, "Collaborative filtering recommender systems," *Found. Trends Hum.-Comput. Interact.*, vol. 4, no. 2, pp. 81–173, Feb. 2010.

[7] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[8] L. Zhang, Z. Chen, M. Zheng, and X. He, "Robust non-negative matrix factorization," *Frontiers Elect. Electron. Eng. China*, vol. 6, no. 2, pp. 192–200, Jun. 2011.

[9] R. Salakhutdinov and A. Mnih, "Probabilistic matrix factorization," in *Proc. Neural Inf. Process. Syst. (NIPS)*, vol. 21, 2008, pp. 1257–1264.

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

[11] C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles," in *Proc. KDD*, vol. 11, 2011, pp. 448–456.

[12] B. Rastegarpanah, M. Crovella, and K. P. Gummadi, "Exploring explanations for matrix factorization recommender systems," in *Proc. FATREC*, no. 1, 2017, pp. 1–4.

[13] D. Barber, *Bayesian Reasoning and Machine Learning*. Cambridge, U.K.: Cambridge Univ. Press, 2012.

[14] M. Ghazanfar and A. Prugel-Bennett, "An improved switching hybrid recommender system using naive Bayes classifier and collaborative filtering," *Computer*, to be published.

[15] J. Bobadilla, A. Gutiérreza, F. Ortega, and B. Zhu, "Reliability quality measures for recommender systems," *Inf. Sci.*, vols. 442–443, pp. 145–157, 2018.

[16] P. Moradi and S. Ahmadian, "A reliability-based recommendation method to improve trust-aware recommender systems," *Expert Syst. Appl.*, vol. 42, no. 21, pp. 7386–7398, Nov. 2015.

[17] S. Kant and T. Mahara, "Merging user and item based collaborative filtering to alleviate data sparsity," *Int. J. Syst. Assur. Eng. Manag.*, vol. 9, no. 1, pp. 173–179, Feb. 2018.

[18] T. Barnard and A. Prügel-Bennett, "Experiments in Bayesian recommendation," in *Advances in Intelligent Web Mastering-3* (Advances in Intelligent and Soft Computing), vol. 86. Berlin, Germany: Springer, 2011, pp. 39–48.

[19] K. Wang and Y. Tan, "A new collaborative filtering recommendation approach based on naive Bayesian method," in *Advances in Swarm Intelligence* (Lecture Notes in Computer Science), vol. 6729. Berlin, Germany: Springer, 2011, pp. 160–167.

[20] F. Ortega, B. Zhu, J. Bobadilla, and A. Hernando, "CF4J: Collaborative filtering for Java," *Knowl.-Based Syst.*, vol. 152, pp. 94–99, Jul. 2018.

[21] F. Ortega, D. Rojo, P. Valdiviezo-Diaz, and L. Raya, "Hybrid collaborative filtering based on users rating behavior," *IEEE Access*, vol. 6, pp. 69582–69591, 2018.

[22] Z. Min and D. Lin, "Probabilistic ensemble of collaborative filters," in *Proc. 32nd AAAI Conf. Artif. Intell. (AAAI)*, 2018, pp. 1–8.

[23] T. B. Nguyen and A. Takasu, "A probabilistic model for the cold-start problem in rating prediction using click data," in *Neural Information Processing* (Lecture Notes in Computer Science), vol. 10638. Cham, Switzerland: Springer, 2017, pp. 196–205.

[24] W. Ma, Y. Wu, M. Gong, C. Qin, and S. Wang, "Local probabilistic matrix factorization for personal recommendation," in *Proc. 13th Int. Conf. Comput. Intell. Secur. (CIS)*, vol. 1, Dec. 2017, pp. 97–101.

[25] T. Wang, C. Rudin, F. Doshi-Velez, Y. Liu, E. Klampfl, and P. MacNeille, "A Bayesian framework for learning rule sets for interpretable classification," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 2357–2393, 2017.

[26] G. Chen, F. Zhu, and P. A. Heng, "Large-scale Bayesian probabilistic matrix factorization with memo-free distributed variational inference," *ACM Trans. Knowl. Discovery Data*, vol. 12, no. 3, pp. 31:1–31:24, 2018.

[27] A. A. Lenskiy and E. Makita, "Bayesian approach to users' perspective on movie genres," *J. Inf. Commun. Converg. Eng.*, vol. 15, no. 1, pp. 43–48, 2017.

[28] A. Hernando, J. Bobadilla, F. Ortega, and A. Gutiérrez, "A probabilistic model for recommending to new cold-start non-registered users," *Inf. Sci.*, vol. 376, pp. 216–232, Jan. 2017.

[29] Y. Weng, L. Wu, and W. Hong, "Bayesian inference via variational approximation for collaborative filtering," *Neural Process. Lett.*, vol. 49, no. 3, pp. 1041–1054 2018.

[30] F. Zafari and I. Moser, "Modelling socially-influenced conditional preferences over feature values in recommender systems based on factorised collaborative filtering," *Expert Syst. Appl.*, vol. 87, pp. 98–117, Nov. 2017.

[31] P. Thakkar, K. Varma, and V. Ukani, "Outcome fusion-based approaches for user-based and item-based collaborative filtering," in *Proc. Inf. Commun. Technol. Intell. Syst. (ICTIS)*, vol. 2, 2017, pp. 127–135.

[32] Y. Shi, H. Lin, and Y. Li, "IU-PMF: Probabilistic matrix factorization model fused with item similarity and user similarity," in *Proc. Int. Conf. Cloud Comput. Secur.*, 2017, pp. 747–758.

[33] R. Chen, Q. Hua, Q. Gao, and Y. Xing, "A hybrid recommender system for Gaussian mixture model and enhanced social matrix factorization technology based on multiple interests," *Math. Problems Eng.*, vol. 2018, Oct. 2018, Art. no. 9109647.

PRISCILA VALDIVIEZO-DIAZ received the Diploma of Advanced Studies (DEA) in Artificial Intelligence from Universidad Nacional de Educación a Distancia, Spain, and the engineering degree in computer science and the master's degree in distance education from the Universidad Técnica Particular de Loja, Ecuador. She is currently pursuing the doctor's degree with the Science and Computer Technologies, Smart Cities at Universidad Politécnica de Madrid. She is currently a Professor with the Computer Science and Electronics Department, Universidad Técnica Particular de Loja. Her research interests include artificial intelligence, recommenser systems, and machine learning.

FERNANDO ORTEGA was born in Madrid, Spain, in 1988. He received the B.S. degree in software engineering, the M.S. degree in artificial intelligence, and the Ph.D. degree in computer sciences from the Universidad Politécnica de Madrid, in 2010, 2011, and 2015, respectively.

From 2008 to 2015, he was a Research Assistant with Intelligent Systems for Social learning and Virtual Environments Research Group. From 2015 to 2017, he was with BigTrueData Leading Machine Learning Projects. From 2017 to 2018, he was an Assistant Professor with the U-tad, Centro Universitario de Tecnología y Arte Digital. Since 2018, he has been an Assistant Professor with the Universidad Politécnica de Madrid. He is author of 30 research papers in most prestigious international journals. He leads several national projects to include machine learning algorithms into the society. His research interests include machine learning, data analysis, and artificial intelligence.

EDUARDO COBOS was born in Madrid, Spain, in 1992. He received the degree in mathematical engineering from the Universidad Complutense de Madrid, in 2017, where he is currently pursuing the master's degree in computational statistical information processing. Since 2017, he works at Ingenio labs in machine learning projects.

RAÚL LARA-CABRERA received the M.Sc. and Ph.D. degrees in computer science from the University of Málaga, Spain, in 2013 and 2015, respectively. He is currently an Assistant professor with the Department of Sistemas Informáticos, Universidad Politécnica de Madrid, Spain. His main research interests include computational intelligence, machine learning, video games, and complex systems.