CHARACTERIZING RELATIONSHIPS THROUGH CO-CLUSTERING
A Probabilistic Approach

Nicola Barbieri, Gianni Costa, Giuseppe Manco and Ettore Ritacco
High Performance Computing and Networking Institute of the Italian National Research Council
v. Pietro Bucci 41C, Arcavacata di Rende (CS), Italy

Keywords: Collaborative filtering, Recommender systems, Block clustering, Co-clustering.

Abstract: In this paper we propose a probabilistic co-clustering approach for pattern discovery in collaborative filtering data. We extend the *Block Mixture Model* in order to learn about the structures and relationships within preference data. The resulting model can simultaneously cluster users into communities and items into categories. Besides its predictive capabilities, the model enables the discovery of significant knowledge patterns, such as the analysis of common trends and relationships between items and users within communities/categories. We reformulate the mathematical model and implement a parameter estimation technique. Next, we show how the model parameters enable pattern discovery tasks, namely: (*i*) to infer topics for each items category and characteristic items for each user community; (*ii*) to model community interests and transitions among topics. Experiments on MovieLens data provide evidence about the effectiveness of the proposed approach.

1 INTRODUCTION

Collaborative Filtering (CF) is recently becoming the dominant approach in *Recommender Systems (RS)*. In literature, several *CF* recommendation techniques have been proposed, mainly focusing on the predictive skills of the system. Recent studies (McNee et al., 2006; Cremonesi et al., 2010) have shown that the focus on prediction does not necessarily helps in devising good recommender systems. Under this perspective, CF models should be considered in a broader sense, for their capability to understand deeper and hidden relationships among users and products they like. Examples in this respect are user communities, item categories preference patterns within such groups. Besides their contribution to the minimization of the prediction error, these relationships are important as they can provide a faithful yet compact description of the data which can be exploited for better decision making.

In this paper we present a co-clustering approach to preference prediction and rating discovery, based on the *Block Mixture Model (BMM)* proposed in (Govaert and Nadif, 2005). Unlike traditional CF approaches, which try to discover similarities between users or items using clustering techniques or matrix decomposition methods, the aim of the BMM is to partition data into homogeneous block enforcing a simultaneous clustering which consider both the dimension of the preference data. This approach highlights the mutual relationship between users and items: similar users are detected by taking into account their ratings on similar items, which in turn are identified considering the ratings assigned by similar users. We extended the original BMM formulation to model each preference observation as the output of a gaussian mixture employing a maximum likelihood (ML) approach to estimate the parameter of the model. Unfortunately, the strict interdependency between user and item cluster makes difficult the application of traditional optimization approaches like EM. Thus, we perform approximated inference based on a variational approach and a two-step application of the EM algorithm which can be thought as a good compromise between the semantic of the original model and the computational complexity of the learning algorithm.

We reformulate standard pattern discovery tasks by showing how a probabilistic block model automatically allows to infer patterns and trends within each block. We show experimentally that the proposed model guarantees a competitive prediction accuracy with regards to standard state-of-the art approaches, and yet it allows to infer topics for each item cate-

Barbieri N., Costa G., Manco G. and Ritacco E..
CHARACTERIZING RELATIONSHIPS THROUGH CO-CLUSTERING - A Probabilistic Approach.
DOI: 10.5220/0003656800640073
In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval* (KDIR-2011), pages 64-73
ISBN: 978-989-8425-79-9

gory, as well as to learn characteristic items for each user community, or to model community interests and transitions among topics of interests. Experiments on both the Netflix and Movielens data show the effectiveness of the proposed model.

2 PRELIMINARIES AND RELATED WORK

User's preferences can be represented by using a $M \times N$ rating matrix $\mathbf{R}$, where M is the cardinality of the *user-set* $\mathcal{U} = \{u_1, \cdots, u_M\}$ and N is the cardinality of the *item-set* $I = \{i_1, \cdots, i_N\}$. The rating value associated to the pair $\langle u, i \rangle$ will be denoted as r_i^u. Typically the number of users and items can be very large, with $M >> N$, and preferences values fall within a fixed integer range $\mathcal{V} = \{1, \cdots, V\}$, where 1 denote the lower interest value. Users tend to express their interest only on a restricted number of items; thus, the rating matrix is characterized by an exceptional sparseness factor (e.g more than 95%). Let $\delta(u,i)$ be a rating-indicator function, which is equals to 1 if the user u has rated/purchased the item i, zero otherwise. Let $I(u)$ denote the set of products rated by the user u: $I(u) = \{i \in I : \delta(u,i) = 1\}$; symmetrically, $\mathcal{U}(i)$ denotes the set of users who have expressed their preference on the item i.

Latent Factor models are the most representative and effective model-based approaches for CF. The underlying assumption is that preference value associated to the pair $\langle u, i \rangle$ can be decomposed considering a set of contributes which represent the interaction between the user and the target item on a set of features. Assuming that there are a set of K features which determine the user's interest on an given item. The assumption is that a rating is the result of the influence of these feature to users and items: $\hat{r}_i^u = \sum_{z=1}^{K} U_{u,z} V_{z,i}$, where $U_{u,z}$ is the response of the user u to the feature z and $V_{z,i}$ is the response on the same feature of the item i.

Several learning schema have been proposed to overcome the sparsity of the original rating matrix and to produce accurate models. The learning phase may be implemented in a deterministic way, via *gradient descent* (Funk, 2006) or, following a probabilistic approach, maximizing the log-likelihood of the model via the *Expectation Maximization* algorithm. The latter leads to the definition of the *Aspect Model*(Hofmann and Puzicha, 1999), known also as *pLSA*. According to the *user community variant*, the rating value r is conditionally independent of the user's identity given her respective community Z; thus, the probability of observing the rating value r for the pair $\langle u, i \rangle$ can be computed as $p(r|u,i) = \sum_{z=1}^{K} p(r|i,z)p(z|u)$, where $P(z|u)$ measures how much the preference values given by u fits with the behavior of the community z and $p(r|i,z)$ is the probability that a user belonging to the community z assigns a rating value r on i.

Only a few co-clustering approaches have been proposed for CF data. An application of the weighted *Bregman coclustering (Scalable CC)* to rating data is discussed in (George and Merugu, 2005). The *two-sided clustering model for CF* (Hofmann and Puzicha, 1999) is based on the strong assumption that each person belongs to exactly one user-community and each item belong to one groups of items, and finally the rating value is independent of the user and item identities given their respective cluster memberships. Let $C = \{c_1, \cdots, c_k\}$ be the user-clusters and let $c(u) : \mathcal{U} \to C$ be a function that maps each user to the respective cluster. Similarly, let $D = \{d_1, \cdots, d_L\}$ be a set of disjoint item-clusters, and $d(i) : I \to D$ is the corresponding mapping function. According to the two-sided clustering model, the probability of observing the preference value r conditioned to the pair $\langle u, i \rangle$ is the following:

$$p(r|u,i,c(u) = c, d(i) = d) = p(r|c,d)$$

where $p(r|c,d)$ are Bernoulli parameters and the cluster membership are estimated by employing a variational inference approach.

The *Flexible Mixture Model (FMM)* (Jin et al., 2006) extends the Aspect and the two sided model, by allowing each user/item to belong to multiple clusters, which are determined simultaneously, according to a coclustering approach. Assuming the existence of K user clusters indexed by c and L item clusters, indexed by d, and let $p(c_k)$ be the probability of observing the user-cluster k with $p(u|c_k)$ being the probability of observing the user profile u given the cluster k and using the same notations for the item-cluster, the joint probability $p(u,i,r)$ is defined as:

$$p(u,i,r) = \sum_{c=1}^{C} \sum_{d=1}^{D} p(c)p(d)p(u|c)p(i|d)p(r|c,d)$$

The predicted rating associated to the pair $\langle u, i \rangle$ is then computed as:

$$\hat{r}_i^u = \sum_{r=1}^{V} r \frac{p(u,i,r)}{\sum_{r'=1}^{V} p(u,i,r')}$$

The major drawback of the FMM relies on the complexity of the training procedure, which is connected with the computation of the probabilities $p(c,d|u,i,r)$ during the Expectation step.
A coclustering extension of the LDA(Blei et al., 2003)

model for rating data have been proposed in (Porteous et al., 2008): the *Bi-LDA* employs two interacting LDA models which enforce the simultaneous clustering of users and items in homogeneous groups.

Other co-clustering approaches have been proposed in the current literature (see (Shan and Banerjee, 2008; Wang et al., 2009)), however their extension to explicit preference data, which requires a distribution over rating values, has not been provided yet.

3 A BLOCK MIXTURE MODEL FOR PREFERENCE DATA

In this section, we are interested in:devising how the available data fits into ad-hoc communities and groups, where groups can involve both users and items. Fig. Fig. 1 shows a toy example of preference data co-clustered into blocks. As we can see, a coclustering induces a natural ordering among rows and columns, and it defines blocks in the rating matrix with similar ratings. The discovery of such a structure is likely to induce information about the population, as well as to improve the personalized recommendations.

Formally, a *block mixture model (BMM)* can be defined by two partitions $(\mathbf{z},\mathbf{w})$ which, in the case of preference data and considering known their respective dimensions, have the following characterizations:

- $\mathbf{z} = z_1,\cdots,z_M$ is a partition of the user set $\mathcal{U}$ into K clusters and $z_{uk} = 1$ if u belongs to the cluster k, zero otherwise;
- $\mathbf{w} = w_1,\cdots,w_N$ is a partition of the item set I into L clusters and $w_{il} = 1$ if the item i belongs to the cluster l, zero otherwise.

Given a rating matrix $\mathbf{R}$, the goal is to determine such partitions and the respective partition functions which specify, for all pairs $\langle u,i\rangle$ the probabilistic degrees of membership wrt. to each user and item cluster, in such a way to maximize the likelihood of the model given the observed data. According to the approach described (Govaert and Nadif, 2005; Gerard and Mohamed, 2003), and assuming that the rating value r observed for the pair $\langle u,i\rangle$ is independent from the user and item identities, fixed z and w, the generative model can be described as follows:

1. For each u generate $z_u \sim Discrete(\pi_1;\ldots;\pi_K)$
2. for each i generate $w_i \sim Discrete(\psi_1;\ldots;\psi_L)$
3. for each pair (u,i):
   - detect k and l such that $z_{uk} = 1$ and $w_{il} = 1$
   - generate $r \sim N(\mu_k^l;\sigma_k^l)$

There are two main differences with respect to the FMM model introduced in the related work. First of all, in our model all cluster membership are assumed given a-priori, whereas FMM models each pair separately. That is, we assume that the cluster memberships z_u and w_i are sampled once and for all, whereas in the FMM model they are sampled for each given pair (u,i). Thus, in the FMM model, a use u can be associated to different clusters in different situations. Although more expressive, this model is prone to overfitting and makes the learning process extremely slow. The second difference is in the way we model the rating probability $p(r|z,w)$. FMM adopts the multinomial model, whereas we choose to adopt the gaussian. The latter better weights the difference between the expected and the observed value: i.e, larger values for $|\hat{r}_i^u - r_i^u|$ introduce a penalty factor.

The corresponding data likelihood in the Block Mixture can be modeled as

$$p(\mathbf{R},\mathbf{z},\mathbf{w}) = \prod_{u\in\mathcal{U}} p(z_u) \prod_{i\in I} p(w_i) \prod_{(u,i,r)\in\mathbf{R}} p(r|z_u,w_i)$$

and consequently, the log-likelihood becomes:

$$\begin{aligned} L_c(\Theta;\mathbf{R},\mathbf{z},\mathbf{w}) = &\sum_{k=1}^{K}\sum_{u\in\mathcal{U}} z_{uk}\log\pi_k + \\ &+\sum_{l=1}^{L}\sum_{i\in I} w_{il}\log\psi_l + \\ &+\sum_{\langle u,i,r\rangle\in\mathbf{R}}\sum_k\sum_l \left[z_{uk}w_{il}\log\varphi(r;\mu_k^l,\sigma_k^l)\right] \end{aligned}$$

where Θ represents the whole set of parameters $\pi_1,\ldots,\pi_K$, $\psi_1,\ldots,\psi_L$, $\mu_1^1,\ldots,\mu_K^L$, $\sigma_1^1,\ldots,\sigma_K^L$ and $\varphi(r;\mu,\sigma)$ is the gaussian density function on the rating value r with parameters μ and σ, i.e., $\varphi(r;\mu;\sigma) = (2\pi)^{-1/2}\sigma^{-1}exp\left(\frac{-1}{2\sigma^2}(r-\mu)^2\right)$.

In the following we show how the model can be inferred and exploited both for prediction and for pattern discovery.

3.1 Inference and Parameter Estimation

Denoting $p(z_{uk}=1|u,\Theta^{(t)}) = c_{uk}$, $p(w_{il}=1|i,\Theta^{(t)}) = d_{il}$ and $p(z_{uk}w_{il}=1|u,i,\Theta^{(t)}) = e_{ukil}$, The conditional expectation of the complete data log-likelihood becomes:

$$\begin{aligned} Q(\Theta;\Theta^{(t)}) \quad = \quad &\sum_{k=1}^{K}\sum_u c_{uk}\log\pi_k + \sum_{l=1}^{L}\sum_i d_{il}\log\psi_l + \\ &\sum_{\langle u,i,r\rangle\in\mathbf{R}}\sum_k\sum_l \left[e_{ukil}\log\varphi(r;\mu_k^l,\sigma_k^l)\right] \end{aligned}$$

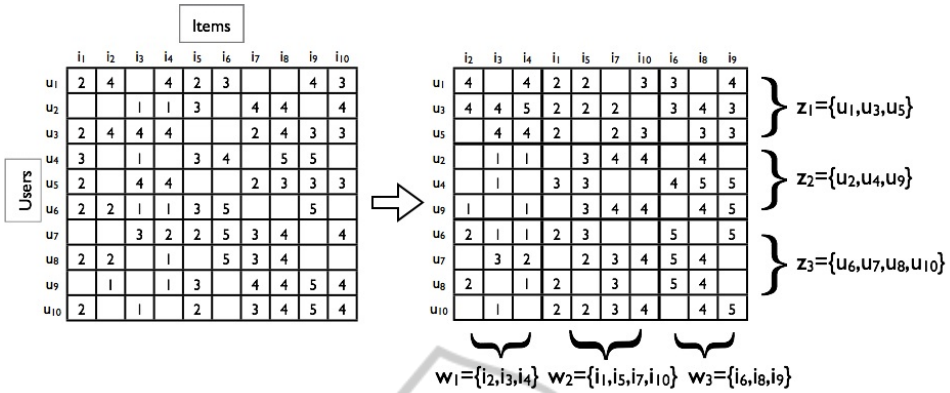


Figure 1: Example Co-Clustering for Preference Data.

As pointed out in (Gerard and Mohamed, 2003), the above function is not tractable analytically, due to the difficulties in determining e_{ukil}; nor the adoption of its variational approximation ($e_{ukil} = c_{uk} \cdot d_{il}$) allows us to derive an Expectation-Maximization procedure for $Q'(\Theta,\Theta^{(t)})$ where the M-step can be computed in closed form. In (Gerard and Mohamed, 2003) the authors propose an optimization of the complete-data log-likelihood based on the *CEM* algorithm. We adapt the whole approach here. First of all, we consider that the joint probability of a a normal population x_i with $i = 1$ to n can be factored as: $\prod_{i=1}^{n} \varphi(x_i;\mu,\sigma) = h(x_1,\ldots,x_n) * \varphi(u_0,u_1,u_2;\mu,\sigma)$, where $h(x_1,\ldots,x_n) = (2\pi)^{-n/2}$, $\varphi(u_0,u_1,u_2;\mu,\sigma) = \sigma^{-u_0} exp\left(\frac{2u_1\mu - u_2 - u_0\mu^2}{2\sigma^2}\right)$ and u_0, u_1 and u_2 are the sufficient statistics.

Based on the above observation, we can define a two-way EM approximation based on the following decompositions of Q':

$$\begin{aligned} Q'(\Theta,\Theta^{(t)}) &= Q'(\Theta,\Theta^{(t)}|\mathbf{d}) + \sum_{i\in I}\sum_{l=1}^{L} d_{il}\log\psi_l \\ &- \sum_{u\in\mathcal{U}}\sum_{i\in I(u)} d_{il}/2\log(2\pi) \end{aligned}$$

where

$$Q'(\Theta,\Theta^{(t)}|\mathbf{d}) = \sum_{u=1}^{M}\sum_{k=1}^{K} c_{uk}\left(\log(\pi_k) + \tau_{uk}\right)$$

$$\tau_{uk} = \sum_{l=1}^{L}\log\left(\varphi(u_0^{(u,l)},u_1^{(u,l)},u_2^{(u,l)};\mu_k^l,\sigma_k^l)\right)$$

$$u_0^{(u,l)} = \sum_{i\in I(u)} d_{il}; \quad u_1^{(u.l)} = \sum_{i\in I(u)} d_{il} r_i^u$$

$$u_2^{(u,l)} = \sum_{i\in I(u)} d_{il}(r_i^u)^2$$

Analogously,

$$\begin{aligned} Q'(\Theta,\Theta^{(t)}) &= Q'(\Theta,\Theta^{(t)}|\mathbf{c}) + \sum_{u\in\mathcal{U}}\sum_{k=1}^{K} c_{uk}\log\pi_k \\ &- \sum_{i\in I}\sum_{u\in\mathcal{U}(i)} c_{uk}/2\log(2\pi) \end{aligned}$$

where

$$Q'(\Theta,\Theta^{(t)}|\mathbf{c}) = \sum_{i=1}^{N}\sum_{l=1}^{L} d_{il}\left(\log(\psi_l) + \tau_{il}\right)$$

$$\tau_{il} = \sum_{k=1}^{K}\log\left(\varphi(u_0^{(i,k)},u_1^{(i.k)},u_2^{(i,k)};\mu_k^l,\sigma_k^l)\right)$$

$$u_0^{(i,k)} = \sum_{u\in I(u)} c_{uk}; \quad u_1^{(i,k)} = \sum_{u\in I(u)} c_{uk} r_i^u$$

$$u_2^{(i,k)} = \sum_{u\in I(u)} c_{uk}(r_i^u)^2$$

The advantage in the above formalization is that we can approach the single components separately and, moreover, for each component it is easier to estimate the parameters. In particular, we can obtain the following:

1. **E-Step (User Clusters):**

$$\begin{aligned} c_{uk} &= p(z_{uk}=1|u) = \frac{p(u|z_k)\cdot\pi_k}{\sum_{k'=1}^{K} p(u|z_{k'})\cdot\pi_{k'}} \\ p(u|z_k) &= \prod_{l=1}^{L}\varphi(u_0^{(u,l)},u_1^{(u.l)},u_2^{(u,l)};\mu_k^l,\sigma_k^l) \end{aligned}$$

2. **M-Step (User Clusters):**

$$\begin{aligned} \pi_k &= \frac{\sum_{u\in\mathcal{U}} c_{uk}}{M} \\ \mu_k^l &= \frac{\sum_{u=1}^{M}\sum_{i\in I(u)} c_{uk} d_{il} r_i^u}{\sum_{u=1}^{M}\sum_{i\in I(u)} c_{uk} d_{il}} \\ (\sigma_k^l)^2 &= \frac{\sum_{u=1}^{M}\sum_{i\in I(u)} c_{uk} d_{il}(r_i^u - \mu_k^l)^2}{\sum_{u=1}^{M}\sum_{i\in I(u)} c_{uk} d_{il}} \end{aligned}$$

3. **E-Step (Item Clusters):**

$$d_{il} = p(w_{il}=1|i) = \frac{p(i|w_l)\cdot\psi_l}{\sum_{l'=1}^{L} p(i|w_{l'})\cdot\psi_{l'}}$$

$$p(i|w_l) = \prod_{k=1}^{K} \varphi(u_0^{(i,k)}, u_1^{(i,k)}, u_2^{(i,k)}; \mu_k^l, \sigma_k^l)$$

4. **M-Step (Item Clusters):**

$$\psi_l = \frac{\sum_{i\in I} d_{il}}{N}$$

$$\mu_k^l = \frac{\sum_{i=1}^{N}\sum_{u\in\mathcal{U}(i)} d_{il}c_{uk}r_i^u}{\sum_{i=1}^{N}\sum_{u\in\mathcal{U}(i)} d_{il}c_{uk}}$$

$$(\sigma_k^l)^2 = \frac{\sum_{i=1}^{N}\sum_{u\in\mathcal{U}(i)} c_{uk}d_{il}(r_i^u-\mu_k^l)^2}{\sum_{i=1}^{N}\sum_{u\in\mathcal{U}(i)} d_{il}c_{uk}}$$

3.2 Rating Prediction

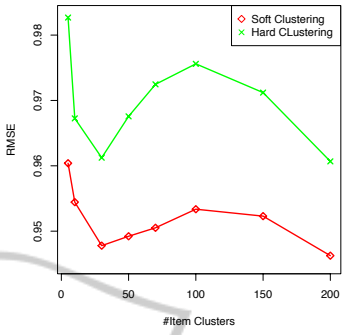
The blocks resulting from a co-clustering can be directly used for prediction. Given a pair $\langle u,i\rangle$, the probability of observing a rating value r associated to the pair $\langle u,i\rangle$ can be computed according to one of the following schemes:

- *Hard-Clustering Prediction*: $p(r|i,u) = \varphi(r;\mu_k^l,\sigma_k^l)$, where $k = \texttt{argmax}_{j=1,\cdots,K} c_{uj}$ and $l = \texttt{argmax}_{h=1,\cdots,L} d_{ih}$ are the clusters that better represent the observed ratings for the considered user and item respectively.
- *Soft-Clustering Prediction*: $p(r|i,u) = \sum_{k=1}^{K}\sum_{l=1}^{L} c_{uk}d_{il}\varphi(r;\mu_k^l,\sigma_k^l)$, which consists of a weighted mixture over user and item clusters.

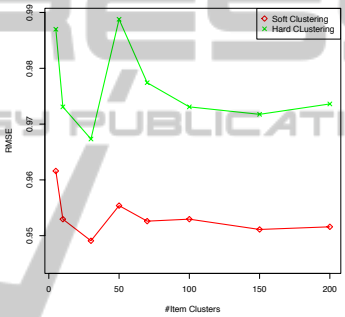
The final rating prediction can be computed by using the expected value of $p(r|u,i)$.

In order to test the predictive accuracy of the BMM we performed a suite of tests on a sample of Netflix data. The training set contains 5,714,427 ratings, given by 435,656 users on a set of 2,961 items (movies). Ratings on those items are within a range 1 to 5 (max preference value) and the sample is 99% sparse. The test set contains 3,773,781 ratings given by a subset of the users (389,305) in the training set over the same set of items. Over 60% of the users have less than 10 ratings and the average number of evaluations given by users is 13.

We evaluated the performance achieved by the BMM considering both the Hard and the Soft prediction rules and performed a suite of experiments varying the number of user and item clusters. Experiments on the three models have been performed by retaining the 10% of the training (user,item,rating) triplets as held-out data and 10 attempts have been executed to determine the best initial configurations. Performance results measured using the RMSE for two BMM with 30 and 50 user clusters are showed in Fig. 2(a) and Fig. 2(b), respectively. In both cases the soft clustering prediction rule overcomes the hard one, and they show almost the same trend. The best result (0.9462) is achieved by employing 30 user clusters and 200 item clusters. We can notice from Tab. 1 that the results follow the same trend as other probabilistic models, like pLSA, which on the same portion of the data achieves 0.9474 accuracy.



(a) 30 user clusters



(b) 50 userclusters

Figure 2: Predictive Accuracy of BMM.

Table 1: RMSE of principal (co-)clustering approaches.

Method	Best RMSE	K	H
BMM	**0.946**	**30**	**200**
PLSA	0.947	30	-
FMM	0.954	10	70
Scalable CC	1.008	10	10

4 PATTERN DISCOVERY USING BMM

The probabilistic formulation of the BMM provides a powerful framework for discovering hidden relation-

ships between users and items. As exposed above, such relationships can have several uses in users segmentation, product catalog analysis, etc. Several works have focused on the application of clustering techniques to discover patterns in data by analyzing user communities or item categories. In (Jin et al., 2004) authors showed how the pLSA model in its co-occurrence version can be used to infer the underlying task of a web browsing session and to discover relationships between users and web pages. Those approaches can be further refined by considering the co-clustering structure proposed so far, which increases the flexibility in modeling both user communities and item categories patterns. Given two different user clusters which group users who have showed a similar preference behavior, the BMM allows the identification of common rated items and categories for which the preference values are different. For example, two user community might agree on action movies while completely disagree on one other. The identification of the topics of interest and their sequential patterns for each user community lead to an improvement of the quality of the recommendation list and provide the user with a more personalized view of the system. In the following we will discuss examples of pattern discovery and user/item profiling tasks.

The experiments in this section were performed considering the 1M MovieLens dataset[1], which contains $1,000,209$ ratings given by $6,040$ users on approximately $3,900$ movies. Each user in this dataset has at least 20 ratings and a list of genres is given for each movie.The latter information will be used to validate the the discovered block structure.

4.1 Co-clustering Analysis

The relationships between groups of users and items captured by the BMM can be easily recognized by analyzing the distribution of the preference values for each cocluster. Given a co-cluster $\langle k,l \rangle$, we can analyze the corresponding distribution of rating values to infer the preference/interest of the users belonging to the community k on item of the category l. Fig. 3 shows graphically a block mixture model with 10 users clusters and 9 item clusters built on the MovieLens dataset. A hard clustering assignment has been performed both on users and clusters: each user u has been assigned to the cluster c such that $c = \texttt{argmax}_{k=1,\cdots,K}\ c_{uk}$. Symmetrically, each item i has been assigned to the cluster d such that: $d = \texttt{argmax}_{l=1,\cdots,L}\ d_{il}$. The background color of each block $\langle k,l \rangle$ describes both the density of ratings and the average preference values given by the users (rows) belonging to the k-th group on items (columns) of the l-th category: the background intensity increases with the average rating values of the co-clusters, which are given in Tab. 2. Each point within the coclusters represents a rating, and again an higher rating value corresponds to a more intense color. The analysis underlines interesting tendencies: for example, users belonging to the user community c_1 tend to assign higher rating values than the average, while items belonging to item category d_6 are the most appreciated. A zoom of portions of the block image is given in Fig. 4(a) and in Fig. 4(b). Here, two blocks are characterized by opposite preference behaviors: the first block contains few (low) ratings, whereas the second block exhibit a higher density of high value ratings.

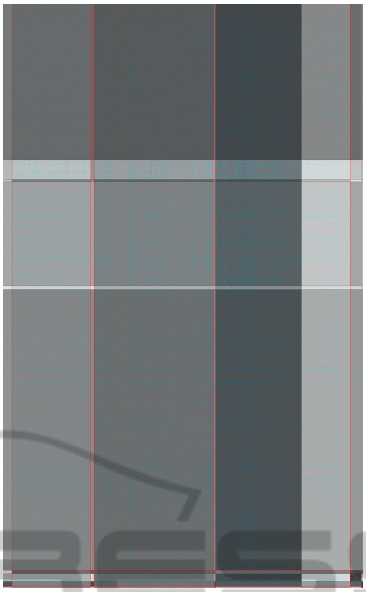


Figure 3: Coclustering.

Table 2: Gaussian Means for each block.

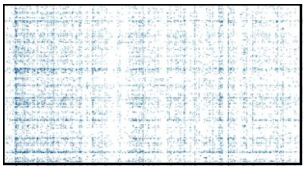
	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9
c_1	3.4	3.59	3.59	4	2.91	4.43	3.59	2.93	3.65
c_2	2.23	2.2	2.92	2.79	2	3.45	2.07	1.80	2.51
c_3	2.11	3.24	3	3.66	2	4.17	1	1.03	5
c_4	2.45	2.69	2.54	3.2	2.43	3.74	2.51	2	2.56
c_5	1	1.79	1	2.32	1	2.98	1.66	1	1.75
c_6	2.93	3.07	3	3.57	2.20	4.09	2.9	2.3	3.16
c_7	1	3.56	3.9	3.7	3.64	3.39	4	3.49	2
c_8	2.25	2.26	1.62	3.27	1	4.17	4.54	1	2.45
c_9	4.08	3.24	4.40	3.54	5	4	3.71	4.5	5
c_{10}	1.91	2.82	1	2.7	4.3	2.2	1	4	2

4.2 Item-topic Analysis

A structural property of interest is the item-topic de-

[1] http://www.grouplens.org/system/files/ml-data-10M100K.tar.gz

(a) Cocluster (c_5, d_8): Avg rating: 1



(b) Cocluster (d_1, d_6): Avg rating: 4.43

Figure 4: Cocluster Analysis.

pendency. Given a set of F topics $\mathcal{G} = \{g_1, \cdots g_F\}$ and assuming that each item is tagged with at least one of those, we can estimate the relevance of each topic within item clusters through a variant of the *tf-idf* measure (Wu et al., 2008), namely *topic frequency - inverse category frequency* (*tf-icf*).

The *topic frequency* (similar to the *term frequency*) of a topic g in a cluster d_l can be defined as:

$$tf_{g,d_l} = \frac{\sum_{i \in d_l} \frac{\delta(g \in Q_i)}{|Q_i|} \sum_{u \in \mathcal{U}} \delta(u,i)}{\sum_{g'=1}^{F} \sum_{i \in d_l} \frac{\delta(g' \in Q_i)}{|Q_i|} \sum_{u \in \mathcal{U}} \delta(u,i)}$$

In a scenario, where items are associated with several topics (genres), and where the number of topics is much lower than size of the itemset, it is high likely that all topics appear at least one in each item category. According to this consideration, the standard definition of *idf* would be useless for our purposes. We, hence, provide an alternative formulation based on entropy (Shannon, 1951), namely *inverse category frequency* (*icf*) for a topic g is:

$$icf_g = 1 + p(g)\log_2[p(g)] + [1-p(g)]\log_2[(1-p(g))]$$

Here, $p(g)$ represent the prior probability of observing a item-genre and is computed as $p(g) = \sum_{l=1}^{L} p(g|d_l) \cdot p(d_l)$, where $p(g|d_l) = tf_{g,d_l}$ and $p(d_l) = \psi_l$.

By combining the above definitions we can finally obtain the *tf-icf* measure for a topic g in a category d_l as:

$$tf\text{-}icf_{g,d_l} = tf_{g,d_l} \times icf_g$$

We can also exploit the fact that BMM provides a soft assignment to clusters, and provide an alternative version of *tf* as:

$$tf_{g,d_l} = \frac{\sum_{i \in d_l} \frac{\delta(g \in Q_i)}{|Q_i)|} \cdot d_{il} \sum_{u \in \mathcal{U}} \delta(u,i)}{\sum_{g'=1}^{F} \sum_{i \in d_l} \frac{\delta(g' \in Q_i)}{|Q_i)|} \cdot d_{il} \sum_{u \in \mathcal{U}} \delta(u,i)}$$

The above considerations can be also applied to the case of item frequency:

$$\begin{aligned} if_{i,d_l} &= \frac{d_{il} \sum_{u \in \mathcal{U}} \delta(u,i)}{\sum_{i' \in d_l} d_{i'l} \sum_{u \in \mathcal{U}} \delta(u,i')} \\ icf_i &= 1 + p(i)\log_2[p(i)] \\ &\quad + [1-p(i)]\log_2[(1-p(i))] \end{aligned}$$

where:

$$p(i) = \frac{|\mathcal{U}(i)|}{|\mathcal{U}|}$$

The topic and item relevance described so far can be directly employed to identify and measure the interest of each user community into topics and items. More specifically, we can measure the interest of a user community c_k for a topic g as:

$$CI_t(c_k, g) = \frac{\sum_{l=1}^{L} \mu_k^l \cdot tf\text{-}icf_{g,d_l}}{\sum_{g'=1}^{F} \sum_{l=1}^{L} \mu_k^l \cdot tf\text{-}icf_{g',d_l}}$$

The item-based counterpart follows straightforwardly:

$$CI_i(c_k, j) = \frac{\sum_{l=1}^{L} \mu_k^l \cdot if\text{-}icf_{j,d_l}}{\sum_{j'=1}^{F} \sum_{l=1}^{L} \mu_k^l \cdot if\text{-}icf_{j',d_l}}$$

where j is the item target.

4.2.1 Evaluation

The MovieLens dataset provides for each movie a list of genres. This information can be used to characterize each item category, by exploiting the within-cluster topic relevance discussed so far. The *tf-icf* measure of observing each genre within each item category is given in Tab. 3, where the dominant topic is in bold.

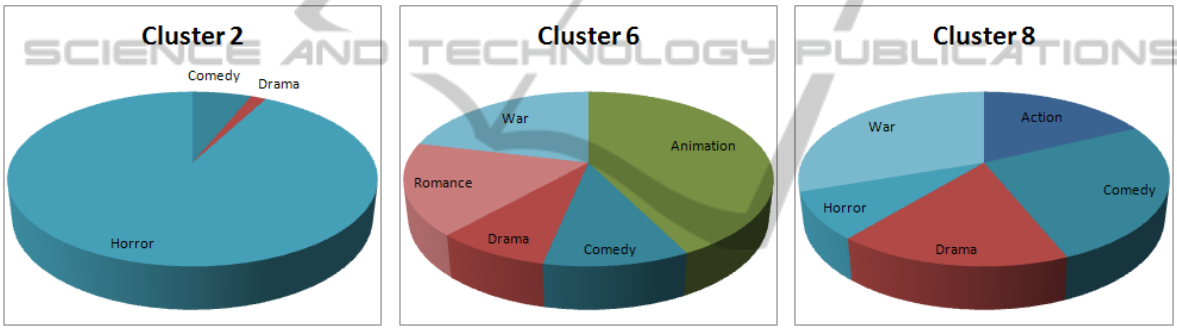
The pie charts in Fig. 5(a), Fig. 5(b) and Fig. 5(c) show the distribution on topics for different item clusters. We can observe different patterns: d_2 is characterized by a strong attitude for horror movies, animation is the dominant topic in cluster 6, and d_8 is summarized by the war genre. Finally, the cluster d_9 shows a predominance of drama movies. A summary of the dominant genres in each item cluster, i.e., with higher *tf-icf*, is given below:

Item Cluster	Dominant Genre
d_1	Drama
d_2	Horror
d_3	Horror
d_4	Action
d_5	Drama
d_6	Animation
d_7	Comedy
d_8	War
d_9	Documentary

Fig. 6 shows the $CI_t(g, c_k)$ values (in gray scale). We can further analyze such values to infer the inter-

Table 3: *tf-icf* measures for each genre in each movie category

	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9
Action	0.03640	0	0.07375	**0.06054**	0.05152	0	0.05624	0.06966	0
Adventure	0.01981	0	0.04237	0.04339	0.03813	0	0.03828	0	0
Animation	0.01591	0	0.00660	0.00926	0.01801	**0.24622**	0.00999	0	0
Children's	0.01581	0	0.03228	0.01643	0.02261	0	0.02855	0	0
Comedy	0.04137	0.03559	0.05403	0.05185	0.04730	0.06209	**0.05685**	0.10228	0
Crime	0.03319	0	0.01585	0.02217	0.01973	0	0.02515	0	0
Documentary	0.01423	0	0.00028	0.00053	0.00291	0	0.00341	0	**0.94466**
Drama	**0.09777**	0.00923	0.02308	0.05247	**0.07720**	0.04839	0.05099	0.06727	0
Fantasy	0.00553	0	0.01175	0.01579	0.01171	0	0.01559	0	0
Film-Noir	0.01485	0	0.00029	0.00123	0.00580	0	0.00113	0	0
Horror	0.01570	**0.53057**	**0.08225**	0.02691	0.01569	0	0.04014	0.03426	0
Musical	0.01739	0	0.00619	0.00914	0.02224	0	0.01088	0	0
Mystery	0.01697	0	0.00832	0.02757	0.00958	0	0.00952	0	0
Romance	0.03470	0	0.02395	0.05776	0.05092	0.09889	0.04625	0	0
Sci-Fi	0.02818	0	0.06247	0.04644	0.03843	0	0.04150	0	0
Thriller	0.04613	0	0.05851	0.05052	0.04771	0	0.05057	0	0
War	0.03902	0	0.01268	0.01041	0.01442	0.12291	0.00716	**0.11860**	0
Western	0.01653	0	0.00625	0.00704	0.00641	0	0.00875	0	0



(a) Item cluster 2 (b) Item cluster 6 (c) Item cluster 8

Figure 5: Topic Analysis on Item Clusters.

est of a user community for a given topic. In particular, a community exhibits a high interest for a topic if the corresponding CI_t value is sufficiently higher than the average CI_t value of all the other topics. Table 4 summarizes the associations among user communities and item topics. For example, users in c_8 exhibit preferences for the *Action* and *War* genres.

4.3 User Profile Segmentation

The topics of interest of a user may change within time and consecutive choices can influence each other. We can analyze such temporal dependencies by mapping each user's choice into their respective item cluster. Assume that movieLens data can be arranged as a set $\{\bar{u}_1,\ldots,\bar{u}_M\}$, where $\bar{u} = \{\langle r_i^u, i, t_i^u\rangle \forall i \in I(u)\}$ and t_i^u is the timestamp corresponding to the rating given by the user u on the item i. By chronologically sorting $\bar{u}$ and segmenting it according to item cluster membership, we can obtain a view of how user's

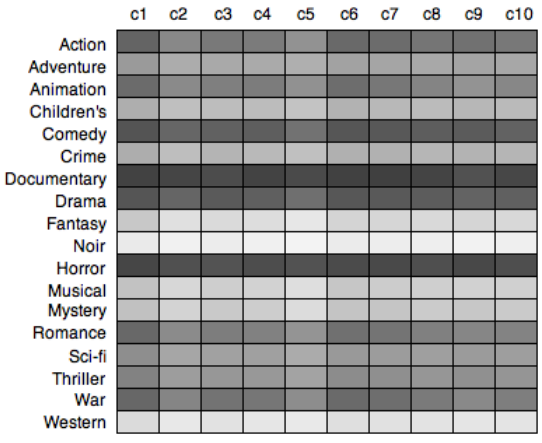


Figure 6: Topic-Interests for User Communities.

tastes change over time. Three example of user profile segmentation are given in the figures below (the mapping between item categories and colors is given

Table 4: Summary of Interests in Topics For User Communities.

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	c_9	c_{10}
Action	y		y					y	y	y
Advent.										
Animat.			y							y
Children's										
Comedy	y	y	y	y	y	y	y	y	y	y
Crime										
Documen.	y	y	y	y	y	y	y	y	y	y
Drama	y	y	y	y	y	y	y	y	y	y
Fantasy										
Noir										
Horror	y	y	y	y	y	y	y	y	y	y
Musical										
Mystery										
Romance			y						y	y
Sci-Fi										
Thriller										
War			y					y		y
Western										

(a)

(b)

(c)

Item Cluster	Color
d_1	Red
d_2	Blue
d_3	Green
d_4	Yellow
d_5	Magenta
d_6	Orange
d_7	Cyan
d_8	Pink
d_9	Dark Grey

Figure 7: User Profile Segmentation.

by the included table).

In practice, we can assume that the three users show a common attitude towards comedy and drama, which are the dominant topics corresponding to the colors yellow and orange. Notice, however, that users (b) and (c) are prone to change their interest towards comedy, as clearly shown by the change in color.

4.4 Modeling Topic Transitions

Based on the above observations, we aim at estimating the sequential connections among topics: In practice, we would like to analyze which item categories are likely to next capture the interests of a user. Those sequential patterns can be modeled by exploiting *Markov Models*. The latter are probabilistic models for discrete processes characterized by the Markov properties. We adopt a Markov Chain property here, i.e., a basic assumption which states that any future state only depends from the present state. This property limits the 'memory' of the chain which can be represented as a digraph where nodes represent the actual states and edges represent the possible transitions among them.

Assuming that the last observed item category for the considered user is d_i, the user could pick an item belonging to the another topic d_j with probability $p(d_j|d_i)$. Thus, we need to estimate all the *transition probabilities*, starting from a $|L+1|$ x $|L+1|$ *transition count matrix* $\mathcal{T}_c$, where $\mathcal{T}_c(i,j)$ stores the number of times that category j follows i in the rating profile of the users.[2]

The estimation we provide is rather simple, corresponding to a simple frequency count:

$$p(d_j|d_i) = \frac{\mathcal{T}_c(i,j)}{\sum_{j=1}^{L+1} \mathcal{T}_c(i,j')}$$

Fig. 8 represents the overall transition probability matrix, which highlights some strong connection among given categories. As instance, the item categories having drama as dominant genre, d_4, d_6 and d_9 are highly correlated as well as d_2, d_7 and d_8 which correspond to comedy movies.

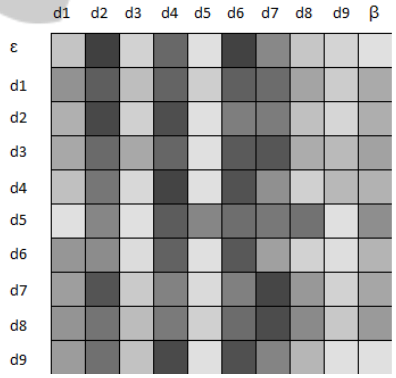
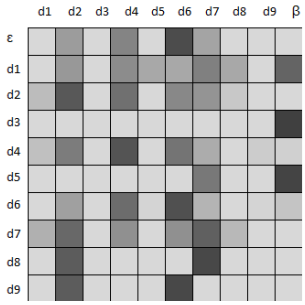


Figure 8: Transition Probabilities Matrix.

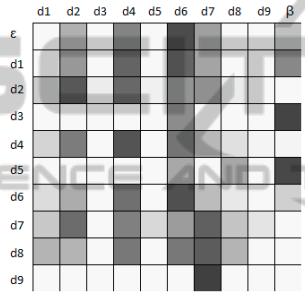
It is interesting to compare how the transition probabilities change within different user communities. Fig. 9 shows the transitions for three different communities. Notice that, besides common transition patterns, each community has some distinctive transitions that characterize their population. For all the considered user communities, the most likely initial item category is d_6; while the first and the last community reproduced in the example show a strong attitude corresponding to the transition $d_8 \rightarrow d_2$, this is

[2]We assume two further states ε, representing the initial choice, and β, representing the last choice.

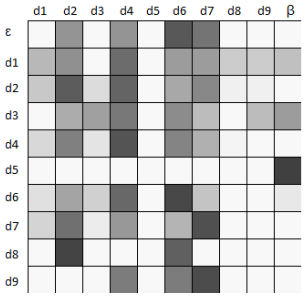
instead a weak pattern within c_7. The same consideration can be done for the transition $d_9 \rightarrow d_7$, which is strong for c_7 and c_{10}, while users belonging to c_3 are more prone to the transition towards d_6.



(a) User community c_3



(b) User community c_7



(c) User community c_{10}

Figure 9: Transition Probabilities Matrix.

The analysis of the transition probabilities can be hence exploited for generating new recommendations enforcing topic diversity (Ziegler et al., 2005) in the top-K lists of items by taking into account not exclusively the current topic of interest but the ones that more likely could be connected to it.

5 CONCLUSIONS AND FUTURE WORKS

In this work we focused on the application of the Block Mixture Model to Collaborative Filtering data. This approach allows the simultaneous clustering of users and items and could be used to identify and measure hidden relationships among them. The proposed model provides a flexible and powerful framework to analyze the users' behavior. This information can be used to improve the quality of a recommendation system, as mentioned throughout the presentation. Future works will focus on embedding baseline components and normalization approaches that might be employed to improve the quality of the clustering and the prediction accuracy.

REFERENCES

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022.

Cremonesi, P., Koren, Y., and Turrin, R. (2010). Performance of recommender algorithms on top-n recommendation tasks. In *RecSys*, pages 39–46.

Funk, S. (2006). Netflix update: Try this at home.

George, T. and Merugu, S. (2005). A scalable collaborative filtering framework based on co-clustering. In *ICDM*, pages 625–628.

Gerard, G. and Mohamed, N. (2003). Clustering with block mixture models. *Pattern Recognition*, 36(2):463–473.

Govaert, G. and Nadif, M. (2005). An em algorithm for the block mixture model. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(4):643–647.

Hofmann, T. and Puzicha, J. (1999). Latent class models for collaborative filtering. In *IJCAI*, pages 688–693.

Jin, R., Si, L., and Zhai, C. (2006). A study of mixture models for collaborative filtering. *Inf. Retr.*, 9(3):357–382.

Jin, X., Zhou, Y., and Mobasher, B. (2004). Web usage mining based on probabilistic latent semantic analysis. In *KDD*, pages 197–205.

McNee, S., Riedl, J., and Konstan, J. A. (2006). Being accurate is not enough: How accuracy metrics have hurt recommender systems. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 1097–1101.

Porteous, I., Bart, E., and Welling, M. (2008). Multi-hdp: a non parametric bayesian model for tensor factorization. In *AAAI*, pages 1487–1490.

Shan, H. and Banerjee, A. (2008). Bayesian co-clustering. In *ICML*.

Shannon, C. E. (1951). Prediction and entropy of printed english. *Bell Systems Technical Journal*, 30:50–64.

Wang, P., Domeniconi, C., and Laskey, K. B. (2009). Latent dirichlet bayesian co-clustering. In *ECML PKDD*, pages 522–537.

Wu, H. C., Luk, R. W. P., Wong, K. F., and Kwok, K. L. (2008). Interpreting tf-idf term weights as making relevance decisions. *ACM Trans. Inf. Syst.*, 26:13:1–13:37.

Ziegler, C.-N., McNee, S. M., Konstan, J. A., and Lausen, G. (2005). Improving recommendation lists through topic diversification. In *WWW*, pages 22–32.