Cross-Domain Collaborative Filtering via Bilinear Multilevel Analysis

Liang Hu[1], Jian Cao[1*], Guandong Xu[2], Jie Wang[3], Zhiping Gu[4], Longbing Cao[2]

[1] Dept of CSE, Shanghai Jiaotong University, {lianghu, cao-jian}@sjtu.edu.cn

[2]Advanced Analytics Institute, University of Technology Sydney, {guandong.xu, longbing.cao}@uts.edu.au

[3]Dept of CEE, Stanford University, jiewang@stanford.edu

[4]Dept of EE, STIEI, guzhiping@stiei.edu.cn

Abstract

Cross-domain collaborative filtering (CDCF), which aims to leverage data from multiple domains to relieve the data sparsity issue, is becoming an emerging research topic in recent years. However, current CDCF methods that mainly consider user and item factors but largely neglect the heterogeneity of domains may lead to improper knowledge transfer issues. To address this problem, we propose a novel CDCF model, the Bilinear Multilevel Analysis (BLMA), which seamlessly introduces multilevel analysis theory to the most successful collaborative filtering method, matrix factorization (MF). Specifically, we employ BLMA to more efficiently address the determinants of ratings from a hierarchical view by jointly considering domain, community, and user effects so as to overcome the issues caused by traditional MF approaches. Moreover, a parallel Gibbs sampler is provided to learn these effects. Finally, experiments conducted on a real-world dataset demonstrate the superiority of the BLMA over other state-of-the-art methods.

1 Introduction

In the era of Web 2.0, searching for what we want and then finding what we really need from a huge amount of online information became a daunting task. As a result, various recommender systems have been proposed to alleviate the information overload problem. As the core component of a recommender system, collaborative filtering (CF) techniques have been widely studied in recent years. In particular, matrix factorization (MF) based latent factor models [Koren, *et al.*, 2009, Weston, *et al.*, 2012] have become dominant in current CF approaches, where MF rates the user *i*'s preference to item *j* according to the interaction of the user-factor vector $\boldsymbol{u}_i$ and the item factor $\boldsymbol{v}_j$ using the inner product, i.e., $R_{ij} = \boldsymbol{u}_i^T \boldsymbol{v}_j$. However, most of the real world data follows the power law, i.e., the majority of users are only associated with a very few items. Therefore, insufficient data has become the major barrier for CF methods including MF, and it leads to two big challenges to the CF: the *cold-start* and *data sparsity* issues [Su and Khoshgoftaar, 2009].

To address these issues, the cross-domain collaborative filtering (CDCF) approach, which leverage data from multiple domains, is becoming an emerging research topic [Li, 2011]. An early neighborhood based CDCF (N-CDCF) method was introduced in [Berkovsky, *et al.*, 2007], but it can easily fail to find similar users or items when data is very sparse. It is possible to directly apply matrix factorization (MF) to CDCF (MF-CDCF) by concatenating the rating matrix of each domain. However, this method may not work properly for the cross-domain scenario. These drawbacks are revealed in an example using two domains *Book* and *Music* as depicted in Figure 1 (a). Empirically, the factors affecting users' preferences for books are different from those affecting users' preferences for music. Therefore, due to such heterogeneities between book factors and music factors, it generally fails to find good estimates for $\boldsymbol{u}_i$ that can simultaneously measure both the preferences for books and music.

Recently, Pan, *et al.* [2010] proposed a transfer learning based MF (TLMF) for CDCF. They assume the existence of an auxiliary domain with dense user data so that knowledge learned in this domain can be transferred to the target domain. However, it is often impossible to find such a dense data domain realistically since the rating matrix is sparse for the majority of users in most domains where the power law pervades. To surmount this barrier, a more reasonable solution is to leverage the complementary data from multiple domains. Collective matrix factorization (CMF) [Singh and Gordon, 2008] is able to leverage data from multiple domains by coupling the rating matrix of each domain along the common *user* dimension. However, all these models may still encounter a special cold-start problem when a user is new to the target domain. As shown in Figure 1 (b), the user-factor vectors for users are co-determined by the feedback in auxiliary and target domains. If no data is available for user *i* in the target domain (marked with a red box), the user-factor vector $\boldsymbol{u}_i$ will have to be simply determined by the user preference data in the auxiliary domain. If such $\boldsymbol{u}_i$ is directly transferred to the target domain and interacts with heterogeneous item factors to measure user preferences, it may yield a poor prediction. We call this issue as the "*blind-transfer*". Recall that the data associated with the majority of users are insufficient and even absent, so the above approaches commonly suffer from such *blind-transfer* issues for realistic online data. For current MF approaches, the rating to an item *j* is fully determined by user factors, i.e., $\boldsymbol{u}_i^T \boldsymbol{v}_j$. Due to this reason, current MF models cannot relieve themselves of the *blind-transfer* issue, especially when the data is extremely sparse.

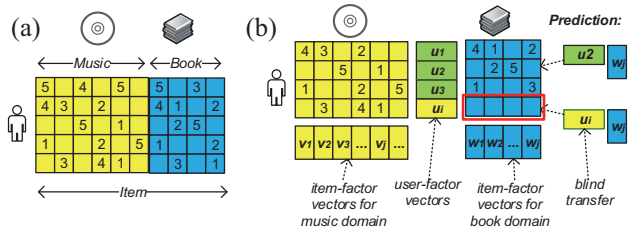

Figure 1: (a) The MF works around the CDCF by concatenating matrices of multiple domains as a single matrix. (b) The demonstration of blind-transfer issue in the CMF and the TLMF.

In this paper, we address the "*blind-transfer*" issue from a new perspective. The basic assumption here is that the ratings for items are determined not only by personal factors but also by *domain* and *community* effects. Let's take an example to illustrate this hypothesis. We argue it is possible to predict a general rating r for *iPhone 5* based only on general market factors, even without any user's feedback. Moreover, the users in different communities often provide biased ratings contrary to each other, for instance, trendspotters may give higher ratings (positive bias) to *iPhone 5* whereas IT engineers may give lower ones (negative bias). Such community-specific bias b_c is associated with the underlying community characteristics. As to a specific user, a personalized bias b_u is given to the general rating r, where b_u can be divided into two parts, $b_{u,g}$ and $b_{u,s}$ repectively. The $b_{u,g}$ is determined by domain-independent user factors that may correlate to personal status, e.g. age, position, income, whereas the $b_{u,s}$ is determined by domain-specific user factors, e.g., a user may prefer expensive phones but cheap clothes. With all that in mind, the personal rating of an item can be expressed as $R = r + b_c + b_{u,g} + b_{u,s}$, i.e., the determinants of a personalized rating is governed by a multilevel factor model apart from flat individual factor models. Hence even when the user data is absent in some domain, $r + b_c + b_{u,g}$ is still available to give an acceptably accurate rating. Therefore, the proposed multilevel factor model is more complete and robust than the traditional individual-only factor models.

To model the above hierarchical factors, we employ the multilevel analysis theory, also referred to as the hierarchical linear model, or the mixed model [Goldstein, *et al.*, 2007, Snijders and Bosker, 2011]. Further, we propose a novel MF model that seamlessly integrates multilevel analysis theory. As MF is a bilinear model, our model is named the Bi-Linear Multilevel Analysis (BLMA).

The main contributions of this paper include:

- We propose an innovative approach to determine the personalized ratings via a multilevel factor model, which is more complete and robust than the classical flat factor models, especially when data is sparse.
- We provide a novel CDCF model, the BLMA, which integrates multilevel analysis theory to the MF model.
- We apply Bayesian inference using parallel Gibbs sampling to learn parameters of the BLMA.
- We empirically evaluate our approach and other state-of-the-art approaches on a real-world data set. The results demonstrate the superiority of our approach.

2 Problem Formulation

Our solution for the CDCF problem, which has a hierarchical structure, is depicted in Figure 2. In this paper, we use d to denote domains, $d \in \{1, \cdots, N_D\}$, and c to index the communities within domain d, $c \in \{1, \cdots, N_C^d\}$. Communities can be defined dedicatedly or discovered from data. Users are indexed by i, for $i \in \{1, \cdots, N_U\}$, and in each domain, a user i only belongs to a community. j is used to index items for each domain, for $j \in \{1, \cdots, N_I^d\}$. The ratings (observed data) are denoted as $\boldsymbol{R} = \{\boldsymbol{R}_{dij} | (d,i,j) \in \boldsymbol{I_R}\}$, where $(d,i,j) \in \boldsymbol{I_R}$ indexes each observation, i.e., the rating that user i gave to item j of domain d. We use $\{(d,i,j) \in \boldsymbol{I_R}(dc)\}$ to denote all indices associated with community c in domain d and similarly we have $\boldsymbol{I_R}(di)$, $\boldsymbol{I_R}(i)$, and etc.

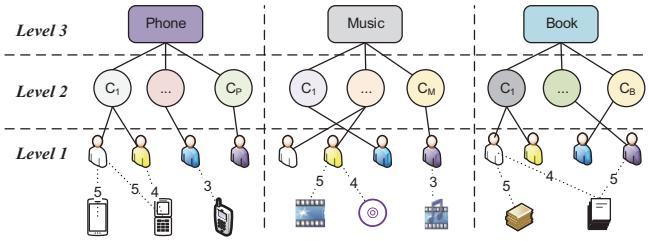

Figure 2: A hierarchical representation of CDCF problem. Users are nested within communities within a domain. Note the same user is marked with the same color across domains.

In Figure 2, it can be observed that users are nested within communities within domains. This is a typical usage scenario of multilevel analysis [Goldstein, *et al.*, 2007, Snijders and Bosker, 2011]. In particular, it is natural to employ the nested random effects model [Rabe-Hesketh and Skrondal, 2008] to analyze such a nested structure as given by Eq. (2), where $\boldsymbol{a}_d$, $\boldsymbol{o}_{dc(i)}$, $\boldsymbol{s}_{di}$ are used to model the nested effects. More specifically, $\boldsymbol{a}_d = [a_1, \cdots, a_K]^T$ represents the fixed effects for domain d (*Level-3*), user i belongs to the community $dc(i)$ (*Level-2*) in domain d, and the random effects of this community are denoted as $\boldsymbol{o}_{dc(i)} \in \mathbb{R}^{K\times 1}$, and $\boldsymbol{s}_{di} \in \mathbb{R}^{K\times 1}$ stands for the domain-specific random effects of user i (*Level-1*).

$$\boldsymbol{R}_{dij} = \mu_d + \boldsymbol{v}_{dj}^T \boldsymbol{u}_{di} + e_{dij} \quad (1)$$

$$\boldsymbol{u}_{di} = \boldsymbol{a}_d + \boldsymbol{o}_{dc(i)} + \boldsymbol{s}_{di} + \boldsymbol{g}_i \quad (2)$$

$$\boldsymbol{o}_{dc(i)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_o^d),\ \boldsymbol{s}_{di} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_s^d),\ \boldsymbol{g}_i \sim \boldsymbol{\mathcal{N}}(\boldsymbol{0}, \boldsymbol{\Sigma}_g) \quad (3)$$

$$\boldsymbol{v}_{dj} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_v^d),\ e_{dij} \sim \mathcal{N}(0, \sigma^2) \quad (4)$$

Moreover, since users are associated with multiple domains in a CDCF problem, $\boldsymbol{g}_i \in \mathbb{R}^{K\times 1}$ in Eq. (2) is used to model such cross-domain user factors. In fact, the $\boldsymbol{g}_i$ has the exact counterpart in multilevel analysis, namely the crossed random effects model [Raudenbush, 1993, Rabe-Hesketh and Skrondal, 2008]. Also, in Eq. (1), μ_d is the mean rating of domain d and $\boldsymbol{v}_{dj} \in \mathbb{R}^{K\times 1}$ stands for the random effects for item j and e_{dij} is the error associated with each observation. Here, the bilinear term $\boldsymbol{v}_{dj}^T \boldsymbol{u}_{di}$ formally represents the multilevel effect over items as discussed previously. Since this model contains both fixed and random effects, it is called a mixed (effects) model in multilevel analysis.

Then, one question that needs to be answered is whether to use fixed or random effects for domain, community and user level respectively. In practice, there is no definite criteria for determining whether fixed or random effects should be used, but some useful discussion can be found in several research papers [Gelman and Hill, 2006, Rabe-Hesketh and Skrondal, 2008]. Here, users can be viewed as a sample from a population since more new users may join, and the number of communities may also change with user growth. So the effects of users and communities are treated as random. In contrast, the effects of a domain do not change with users, i.e., the same for all users, so it is modeled as fixed effects.

3 Bayesian Bilinear Multilevel Analysis

3.1 Model

For the model given by Eq. (1), if we treat $\boldsymbol{v}_{dj}$ as a known vector to serve as the covariates, then Eq. (1) is reduced to a linear mixed model (LMM). Alternatively, if $\boldsymbol{u}_{di}$ is treated as known covariates, Eq. (1) becomes a linear random effects model. Now let us move μ_d from the right side to the left side of Eq. (1), so the reformed equation is given by:

$$\boldsymbol{Y}_{dij} = \boldsymbol{v}_{dj}^T \boldsymbol{u}_{di} + e_{dij}, \text{ where } \boldsymbol{Y}_{dij} = \boldsymbol{R}_{dij} - \mu_d \qquad (5)$$

Immediately, we find that the mixed effects $\boldsymbol{u}_{di}$ and the random effects $\boldsymbol{v}_{dj}$ in Eq. (5) exactly correspond to K dimensional user and item factor vector in the MF. That is, we can seamlessly bridge multilevel analysis to the MF. Hence we call Eq. (1) the Bilinear Multilevel Analysis (BLMA), because the bilinear terms $\boldsymbol{u}_{di}$ and $\boldsymbol{v}_{dj}$ are unknown and needs to be learned as classical MF models.

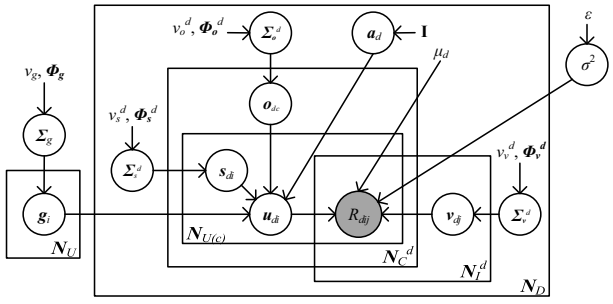

Figure 3: The graphical model for BLMA

In this paper, we formalize the BLMA using a Bayesian probabilistic model. The corresponding graphical representation of this model is illustrated in Figure 3. Accordingly, the conditional distribution over the observed data $\boldsymbol{R}$ (cf. Eq. (1)) can be easily written as:

$$p(\boldsymbol{R}|\boldsymbol{v}, \boldsymbol{u}, \sigma^2) = \prod_{(d,i,j)\in \boldsymbol{I}_R} \mathcal{N}\left(\mu_d + \boldsymbol{v}_{dj}^T \boldsymbol{u}_{di}, \sigma^2\right) \qquad (6)$$

For Bayesian analysis of multilevel models, broadly speaking, two types of prior distributions are available: (a) uninformative; (b) informative [Browne, 1998, Goldstein, *et al.*, 2007]. The uniform prior is a commonly used uninformative prior for fixed effects, i.e., $p(\boldsymbol{a}_d) \propto \mathbf{I}$ for the domain effects $\boldsymbol{a}_d$. In practice, it can be approximated as $p(\boldsymbol{a}_d) \sim \mathcal{N}(\mathbf{0}, c\mathbf{I})$, where c is very large. For random effects, informative priors should be placed on variance parameters. The inverse gamma prior, $\Gamma^{-1}(\varepsilon, \varepsilon)$, with small ε, is often placed on scalar variance while the inverse Wishart prior, $\mathcal{W}^{-1}(\nu, \boldsymbol{\Phi})$, is used for variance matrix [Browne, 1998, Browne and Draper, 2006]. In this paper, without loss of generality, we set $\varepsilon = 0.001$ for $\Gamma^{-1}(\varepsilon, \varepsilon)$ and $\nu = \mathrm{K}, \boldsymbol{\Phi} = \mathbf{I}$ for all $\mathcal{W}^{-1}(\nu, \boldsymbol{\Phi})$. Then, the priors for the variance parameters w.r.t. the random effects, $\boldsymbol{o}_{dc}$, $\boldsymbol{s}_{di}$, $\boldsymbol{g}_i$, $\boldsymbol{v}_{dj}$ and error e_{dij}, can be given as follows:

$$\begin{aligned} & p(\boldsymbol{\Sigma}_{\boldsymbol{o}}^d) = \mathcal{W}^{-1}(\nu_o^d, \boldsymbol{\Phi}_{\boldsymbol{o}}^d),\ p(\boldsymbol{\Sigma}_{\boldsymbol{s}}^d) = \mathcal{W}^{-1}(\nu_s^d, \boldsymbol{\Phi}_{\boldsymbol{s}}^d) \\ & p(\boldsymbol{\Sigma}_{\boldsymbol{g}}) = \mathcal{W}^{-1}(\nu_g, \boldsymbol{\Phi}_{\boldsymbol{g}}),\ p(\boldsymbol{\Sigma}_{\boldsymbol{v}}^d) = \mathcal{W}^{-1}(\nu_v^d, \boldsymbol{\Phi}_{\boldsymbol{v}}^d) \\ & p(\sigma^2) = \Gamma^{-1}(\varepsilon, \varepsilon) \end{aligned} \qquad (7)$$

Obviously, this model lends itself to a full Bayesian analysis by a Markov Chain Monte Carlo (MCMC) method.

3.2 Inference

MCMC methods aim to generate samples from a joint posterior $p(\boldsymbol{\Theta}|\boldsymbol{Y})$ of all unknown parameters $\boldsymbol{\Theta}$. In the BLMA, $\boldsymbol{\Theta} = \{\boldsymbol{a}, \boldsymbol{o}, \boldsymbol{s}, \boldsymbol{g}, \boldsymbol{v}, \boldsymbol{\Sigma}_{\boldsymbol{o}}, \boldsymbol{\Sigma}_{\boldsymbol{s}}, \boldsymbol{\Sigma}_{\boldsymbol{v}}, \sigma^2\}$ represents fixed/random effects and variances. Directly sampling all parameters in $\boldsymbol{\Theta}$ from the joint posterior is intractable in the BLMA because the product of two random variables, $\boldsymbol{v}_{dj}^T \boldsymbol{u}_{di}$, does not belong to any distribution in general. Fortunately, the Gibbs sampler can approximate the joint posterior by sampling each variable in $\boldsymbol{\Theta}$ in turn, conditioned on other variables with current values. Best of all, the Gibbs sampling algorithm for the BLMA is nearly identical to that for the LMM, since the BLMA is reduced to the LMM when sampling $\boldsymbol{u}$ conditioning on $\boldsymbol{v}$, and vice versa.

Referring to the Gibbs samplers for the LMM [Browne, 1998, Browne and Draper, 2000, Goldstein, *et al.*, 2007], we design a parallel sampling algorithm for the BLMA below:

Algorithm 1: Parallel Gibbs sampling scheme for the BLMA

- Draw samples for $\boldsymbol{\Psi} = \{\boldsymbol{a}, \boldsymbol{o}, \boldsymbol{s}, \boldsymbol{g}, \boldsymbol{\Sigma}_{\boldsymbol{o}}, \boldsymbol{\Sigma}_{\boldsymbol{s}}, \boldsymbol{\Sigma}_{\boldsymbol{g}}, \sigma^2\}$ as LMM when $\boldsymbol{v}$ is given and acts as the covariates.
  1. For $d \in \{1, \cdots, N_D\}$, sample the domain effects $\boldsymbol{a}_d$ in parallel:

$$\boldsymbol{a}_d|\boldsymbol{v}, \boldsymbol{\Psi} \backslash \boldsymbol{a}_d \sim \mathcal{N}\left(\widehat{\boldsymbol{a}}_d, \widehat{\boldsymbol{\Sigma}}_d\right) \qquad (8)$$

where $\widehat{\boldsymbol{\Sigma}}_d = \sigma^2 \left(\sum_{(d,i,j)\in \boldsymbol{I}_R(d)} \boldsymbol{v}_{dj} \boldsymbol{v}_{dj}^T\right)^{-1}$

$$\widehat{\boldsymbol{a}}_d = \widehat{\boldsymbol{\Sigma}}_d \left[\sum_{(d,i,j)\in \boldsymbol{I}_R(d)} \frac{\boldsymbol{v}_{dj}\left[\boldsymbol{R}_{dij} - \mu_d - \boldsymbol{v}_{dj}^T(\boldsymbol{o}_{dc(i)} + \boldsymbol{s}_{di} + \boldsymbol{g}_i)\right]}{\sigma^2}\right]$$

  2. For $c \in \{1, \cdots, N_C^d\}$, sample the community effects $\boldsymbol{o}_{dc}$ in parallel for each domain d:

$$\boldsymbol{o}_{dc}|\boldsymbol{v}, \boldsymbol{\Psi} \backslash \boldsymbol{o}_{dc} \sim \mathcal{N}\left(\widehat{\boldsymbol{o}}_{dc}, \widehat{\boldsymbol{\Sigma}}_{dc}\right) \qquad (9)$$

where $\widehat{\boldsymbol{\Sigma}}_{dc} = \left(\sum_{(d,i,j)\in \boldsymbol{I}_R(dc)} \frac{\boldsymbol{v}_{dj} \boldsymbol{v}_{dj}^T}{\sigma^2} + \boldsymbol{\Sigma}_{\boldsymbol{o}}^{d\,-1}\right)^{-1}$

$$\widehat{\boldsymbol{o}}_{dc} = \widehat{\boldsymbol{\Sigma}}_{dc} \left[\sum_{(d,i,j)\in \boldsymbol{I}_R(dc)} \frac{\boldsymbol{v}_{dj}\left[\boldsymbol{R}_{dij} - \mu_d - \boldsymbol{v}_j^T(\boldsymbol{a}_d + \boldsymbol{s}_{di} + \boldsymbol{g}_i)\right]}{\sigma^2}\right]$$

  3. For $i \in \{1, \cdots, N_U\}$, sample the global user effects $\boldsymbol{g}_i$ in parallel:

$$\boldsymbol{g}_i|\boldsymbol{v}, \boldsymbol{\Psi} \backslash \boldsymbol{g}_i \sim \mathcal{N}\left(\widehat{\boldsymbol{g}}_i, \widehat{\boldsymbol{\Sigma}}_i\right) \qquad (11)$$

where $\widehat{\boldsymbol{\Sigma}}_i = \left(\sum_{(d,i,j)\in \boldsymbol{I_R}(i)} \frac{\boldsymbol{v}_{dj}\boldsymbol{v}_{dj}^T}{\sigma^2} + \boldsymbol{\Sigma}_{\boldsymbol{g}}^{-1}\right)^{-1}$

$\widehat{\boldsymbol{g}}_i = \widehat{\boldsymbol{\Sigma}}_i \left[\sum_{(d,i,j)\in \boldsymbol{I_R}(i)} \frac{\boldsymbol{v}_{dj}\left[R_{dij} - \mu_d - \boldsymbol{v}_{dj}^T(\boldsymbol{a}_d + \boldsymbol{o}_{dc(i)} + \boldsymbol{s}_{di})\right]}{\sigma^2}\right]$

4. For $i \in \{1, \cdots, N_U\}$, sample the domain-specific user effects $\boldsymbol{s}_{di}$ in parallel for each domain d:

$$\boldsymbol{s}_{di}|\boldsymbol{v}, \boldsymbol{\Psi}\backslash\boldsymbol{s}_{di} \sim \mathcal{N}\left(\widehat{\boldsymbol{s}}_{di}, \widehat{\boldsymbol{\Sigma}}_{di}\right) \quad (10)$$

where $\widehat{\boldsymbol{\Sigma}}_{di} = \left(\sum_{(d,i,j)\in \boldsymbol{I_R}(di)} \frac{\boldsymbol{v}_{dj}\boldsymbol{v}_{dj}^T}{\sigma^2} + \boldsymbol{\Sigma}_{\boldsymbol{s}}^{d\,-1}\right)^{-1}$

$\widehat{\boldsymbol{s}}_{di} = \widehat{\boldsymbol{\Sigma}}_{di} \left[\sum_{(d,i,j)\in \boldsymbol{I_R}(di)} \frac{\boldsymbol{v}_{dj}\left[R_{dij} - \mu_d - \boldsymbol{v}_{dj}^T(\boldsymbol{a}_d + \boldsymbol{o}_{dc(i)} + \boldsymbol{g}_i)\right]}{\sigma^2}\right]$

5. Sample variance parameters, $\boldsymbol{\Sigma_o}, \boldsymbol{\Sigma_s}, \boldsymbol{\Sigma_g}, \sigma^2$:

$\boldsymbol{\Sigma}_{\boldsymbol{o}}^d|\boldsymbol{\Psi}\backslash\boldsymbol{\Sigma}_{\boldsymbol{o}}^d \sim \mathcal{W}^{-1}(\nu_o^d + N_C^d, \boldsymbol{\Phi}_{\boldsymbol{o}}^d + \sum_c \boldsymbol{o}_{dc}\boldsymbol{o}_{dc}^T)$

$\boldsymbol{\Sigma}_{\boldsymbol{s}}^d|\boldsymbol{\Psi}\backslash\boldsymbol{\Sigma}_{\boldsymbol{s}}^d \sim \mathcal{W}^{-1}(\nu_s^d + N_U, \boldsymbol{\Phi}_{\boldsymbol{s}}^d + \sum_i \boldsymbol{s}_{di}\boldsymbol{s}_{di}^T)$

$\boldsymbol{\Sigma_g}|\boldsymbol{\Psi}\backslash\boldsymbol{\Sigma_g} \sim \mathcal{W}^{-1}(\nu_g + N_U, \boldsymbol{\Phi_g} + \sum_i \boldsymbol{g}_i\boldsymbol{g}_i^T)$

$\sigma^2|\boldsymbol{\Psi}\backslash\sigma^2 \sim \Gamma^{-1}(\varepsilon + |\boldsymbol{R}|/2, \varepsilon + \sum_{d,i,j\in \boldsymbol{I_R}} e_{dij}^2/2)$

where $e_{dij} = \boldsymbol{R}_{dij} - \mu_d - \boldsymbol{v}_{dj}^T\boldsymbol{u}_{di}$

- Draw samples for $\boldsymbol{\Psi} = \{\boldsymbol{v}, \boldsymbol{\Sigma}_{\boldsymbol{v}}, \sigma^2\}$ as the LMM when $\boldsymbol{u}$ is given and acts as the covariates.

1. For $j \in \{1, \cdots, N_C^d\}$, sample the item effects $\boldsymbol{v}_{dj}$ in parallel for each domain d:

$$\boldsymbol{v}_{dj}|\boldsymbol{u}, \boldsymbol{\Psi}\backslash\boldsymbol{v}_{dj} \sim \mathcal{N}\left(\widehat{\boldsymbol{v}}_{dj}, \widehat{\boldsymbol{\Sigma}}_{dj}\right) \quad (12)$$

where $\widehat{\boldsymbol{\Sigma}}_{dj} = \left(\sum_{(d,i,j)\in \boldsymbol{I_R}(d)} \frac{\boldsymbol{u}_{di}\boldsymbol{u}_{di}^T}{\sigma^2} + \boldsymbol{\Sigma}_{\boldsymbol{v}}^{d\,-1}\right)^{-1}$

$\widehat{\boldsymbol{v}}_{dj} = \widehat{\boldsymbol{\Sigma}}_{dj} \left[\sum_{(d,i,j)\in \boldsymbol{I_R}(d)} \frac{\boldsymbol{u}_{di}\left[\boldsymbol{R}_{dij} - \mu_d\right]}{\sigma^2}\right]$

2. Sample variance parameters, $\boldsymbol{\Sigma}_{\boldsymbol{v}}^d, \sigma^2$:

$\boldsymbol{\Sigma}_{\boldsymbol{v}}^d|\boldsymbol{\Psi}\backslash\boldsymbol{\Sigma}_{\boldsymbol{v}} \sim \mathcal{W}^{-1}(\nu_v + N_I^d, \boldsymbol{\Phi_v} + \sum_j \boldsymbol{v}_{dj}\boldsymbol{v}_{dj}^T)$

$\sigma^2|\boldsymbol{\Psi}\backslash\sigma^2 \sim \Gamma^{-1}(\varepsilon + |\boldsymbol{R}|/2, \varepsilon + \sum_{(d,i,j)\in \boldsymbol{I_R}} e_{dij}^2/2)$

Note that appropriate initial values can fairly speed up the convergence of MCMC methods. For the BLMA, we may use a MF method, such as the BPMF [Salakhutdinov and Mnih, 2008], to generate initial values, $\boldsymbol{v}_d$ and $\boldsymbol{\Sigma}_{\boldsymbol{v}}^d$, w.r.t. items for each domain d. Then, using $\boldsymbol{v}$ as the covariates for the LMM, we can initialize $\boldsymbol{a}, \boldsymbol{o}, \boldsymbol{s}, \boldsymbol{g}, \boldsymbol{\Sigma_o}, \boldsymbol{\Sigma_s}, \sigma^2$ by the likelihood estimates [Browne and Draper, 2006, Goldstein, *et al.*, 2007].

3.3 Prediction

After the burn-in period, MCMC methods can predict missing ratings, $\widehat{\boldsymbol{R}}_{dij}$, in terms of the predictive posterior mean, $\mathbb{E}\left(\widehat{\boldsymbol{R}}_{dij}|\boldsymbol{R}\right)$. This mean is often computed by the Monte Carlo approximation from S samples:

$$\mathbb{E}\left(\widehat{\boldsymbol{R}}_{dij}|\boldsymbol{R}\right) \approx \frac{1}{S}\sum_{s=1}^{S} \mathbb{E}\left(\widehat{\boldsymbol{R}}_{dij}|\boldsymbol{u}_{di}^{(s)}, \boldsymbol{v}_{dj}^{(s)}, \boldsymbol{R}\right)$$

$$\mathbb{E}\left(\widehat{\boldsymbol{R}}_{dij}|\boldsymbol{u}_{di}^{(s)}, \boldsymbol{v}_{dj}^{(s)}, \boldsymbol{R}\right) = \mu_d + \boldsymbol{v}_{dj}^{(s)T}\boldsymbol{u}_{di}^{(s)} \quad (13)$$

where the mean w.r.t. each sample is obtained according to Eq. (6), and the effects $\boldsymbol{u}_{di}^{(s)} = \boldsymbol{a}_d^{(s)} + \boldsymbol{o}_{\boldsymbol{dc}(i)}^{(s)} + \boldsymbol{s}_{di}^{(s)} + \boldsymbol{g}_i^{(s)}$ and $\boldsymbol{v}_{dj}^{(s)}$ are sampled from the Markov chain by Eq. (8) ~ (12).

Noted that the domain-specific effects $\hat{\boldsymbol{s}}_{di}^{(s)}$ do not exist if user i is new in domain d, we can simply let it be the population mean of user factors, that is $\mathbf{0}$ (cf. Eq. (3)), or the community sample mean, namely $\sum_{k\in \boldsymbol{C}_d(i)} \hat{\boldsymbol{s}}_{dk}^{(s)} / |\boldsymbol{C}_d(i)|$, where $\boldsymbol{C}_d(i)$ returns all non-cold-start users from i's community.

3.4 Discussions

So far we have presented the prediction method for the BLMA, now we can formally explain why the BLMA can work and how it avoids the *blind-transfer* issue in the CDCF.

In the BLMA, the domain effects $\widehat{\boldsymbol{a}}_d$ is regressed from the data for all items (cf. Eq. (8)) in a domain, so $r_{dij} = \mu_d + \boldsymbol{v}_{dj}^T\boldsymbol{a}_d$, gives the general rating to item j, which is fixed for all users. The community effects $\widehat{\boldsymbol{o}}_{dc}$ is estimated according to the ratings given by the users in this community (cf. Eq. (9)), so $b_{cij} = \boldsymbol{v}_{dj}^T\boldsymbol{o}_{dc}$ provides the community-specific bias. As to the user level, $\widehat{\boldsymbol{g}}_i$ is estimated from user's feedbacks over all domains (cf. Eq. (11)), so $b_{dij}^g = \boldsymbol{v}_{dj}^T\boldsymbol{g}_i$ gives the bias based on a user's global preference. $\hat{\boldsymbol{s}}_{di}$ is estimated from user's feedbacks in a domain (cf. Eq. (10)), so $b_{dij}^s = \boldsymbol{v}_{dj}^T\boldsymbol{s}_{di}$ gives the bias according to domain-specific preferences. Hence $\widehat{\boldsymbol{R}}_{dij} = r_{dij} + b_{cij} + b_{dij}^s + b_{dij}^g$ provides a multilevel effects based rating predictor as given by Eq. (13).

It has been demonstrated that leveraging social links [Ma, *et al.*, 2008] or neighborhood data [Koren, 2008] can significantly improve prediction performance. Here, the BLMA goes a step further, it models not only the community effects to correlate users but also the other effects of domains and individuals to discover similarities. Such multilevel effects are particularly useful when individual user data is insufficient. If we remove the domain effects $\boldsymbol{a}_d$, community effects $\boldsymbol{o}_{dc}$, and domain-specific user effects $\boldsymbol{s}_{di}$ from Eq. (2), the BLMA degenerates to the CMF where only the flat single-level user factors $\boldsymbol{g}_i$ are modeled. Obviously, simply using $\boldsymbol{g}_i$ is vulnerable to the heterogeneity issue as discussed at the beginning. In comparison, the domain effects $\boldsymbol{a}_d$ and the community effects $\boldsymbol{o}_{dc}$ enable the BLMA to more efficiently tackle issues of domain heterogeneity and data sparsity.

Another point of concern is the membership of community. These memberships can be directly assigned according to real-world memberships. For example, fan clubs for cars and groups for games are two types of easily retrieved memberships from two different domains. If such ready-made memberships are not available, we can discover them from data [Fortunato, 2010]. In this paper, we only consider a user belonging to a single community within a domain, but the multiple membership can also be dealt with easily by assigning different weights [Browne, *et al.*, 2001].

4 Related Work

We propose to solve the CDCF problem by using the BLMA, which can be regarded as an approach that couples multilevel analysis with MF. Hence, the applications of multilevel analysis and MF based approaches for CDCF are the two most relevant research areas.

Codebook Transfer [Li, *et al.*, 2009] assumes some cluster-level rating patterns, which are represented by a codebook,

can be found between the rating matrices in two related domains. The Rating-Matrix Generative Model [Li, *et al.*, 2009] extends this idea with a probabilistic model to solve collective transfer learning problems. In reality, there are many cold-start users for most domains. Therefore, it is always out of the question to seek common patterns among domains without user data. The Coordinate System Transfer [Pan, *et al.*, 2010] learns the user-factor matrix $\mathbf{U}_A$ from an auxiliary rating matrix in the first step, and then generates the user-factor matrix $\mathbf{U}_T$ for the target domain based on $\mathbf{U}_A$, with the regularization of penalizing the divergence between $\mathbf{U}_A$ and $\mathbf{U}_T$. As pointed out at the beginning, this approach will run into the blind-transfer issue for cold-start users. Therefore, all the above approaches are not applicable to the CDCF problem over multiple domains as studied in this paper.

Although the CMF [Singh and Gordon, 2008] can address the CDCF problem over multiple domains by coupling the common user dimension, it still suffers from the blind-transfer issue. Moreover, CMF uses a set of weights to trade off the loss of the fitting rating matrix from each domain. However, searching an optimal combination of weights to tune the prediction performance is a heuristic and computationally expensive task. In comparison, the BLMA applies Bayesian analysis to assign priors on all variables and learns them by Gibbs sampling. In fact, we have shown that the BLMA can be reduced to the flat Bayesian CMF [Singh and Gordon, 2012] by removing all high level group effects.

The multilevel analysis [Snijders and Bosker, 2011] is a popular statistic model theory, which has been widely studied and applied to many areas, including economics, education, sociology, biology, health, and beyond [Raudenbush, 1993, Gelman and Hill, 2006, Goldstein, *et al.*, 2007, Rabe-Hesketh and Skrondal, 2008]. The BLMA brings multilevel analysis from the traditional linear world to the bilinear world and brings multilevel effects to the MF instead of the traditional individual factors. Being an integrated approach with more robust models for tackling complex data sets, the BLMA is proposed for applications beyond the CDCF problem.

5 Experiments

The experiments were conducted on a real-world dataset, that is, the ratings for Amazon products. We evaluated the prediction performance using the BLMA and other state-of-the-art approaches to demonstrate the superiority of our model.

Table 1: Statistics of Amazon dataset for evaluation

Domain	# Items	Density	avg. # ratings/user
Book	6000	0.0097	57
Music	5000	0.0062	30
DVD	3000	0.0124	37
VHS	3000	0.0117	35

5.1 Data Preparation

The dataset was crawled from the publicly available Amazon website, where it contains 1,555,170 users and 1-5 scaled ratings over 548,552 different products covering four domains: 393,558 books, 103,144 music CDs, 19,828 DVDs and 26,132 VHS video tapes [Leskovec, *et al.*, 2007]. Obviously, these domains are different but there are some common user factors to affect preferences across these domains, so this dataset is highly suitable for testing the CDCF algorithms.

We filtered out users who have rated at least 50 books or 30 music CDs so that there are enough observations to be split in various proportions of training and testing data for our evaluation. Finally, 2,505 users were selected, and in addition we retrieved all items rated by these users in these four domains and set aside top K rated items for each domain respectively. We use $\boldsymbol{\mathcal{D}}$ to denote this extracted data set. Table 1 shows the statistics of $\boldsymbol{\mathcal{D}}$. In this experiment, we evaluated the prediction performance by using *Book* and *Music CD* as the testing domains respectively. Accordingly, we constructed a set of different training/testing sets as follows.

- *Sparse data cases:* We constructed two different sparsities of training sets by respectively holding out 80% and 25% of ratings for the *Book* domain from $\boldsymbol{\mathcal{D}}$, when testing the prediction over books. The hold-out data serves as the ground truth for testing. Likewise, we constructed two other training sets when testing the prediction over music CDs by respectively holding out 80% and 25% of ratings for the *Music CD* domain from $\boldsymbol{\mathcal{D}}$.
- *Cold-start cases*: To simulate the cold-start cases, we constructed two training sets by randomly selecting half of users and holding out all their data from the *Book* and the *Music CD* domains respectively.

5.2 Metrics and Comparative Methods

The *Mean Absolute Error* (MAE) is the most widely used evaluation metric to measure the prediction quality for collaborative filtering. In Eq. (14), $r_{i,j}$ denotes the true rating user i gave to item j, $\hat{r}_{i,j}$ is the predicted rating, and $N = |\boldsymbol{T}|$ is the number of ratings in the testing set.

$$\text{MAE} = \sum_{r_{i,j} \in \boldsymbol{T}} ABS\left(r_{i,j} - \hat{r}_{i,j}\right)/N \qquad (14)$$

In this experiment, a group of state-of-the-art methods for the CDCF problem are given by:

- *MF-SGD:* The most well-known MF method for single domain CF by minimizing the squared error by stochastic gradient descent [Koren, *et al.*, 2009]. It is simply run over the rating matrix of the testing domain.
- *N-CDCF-U:* The user-based neighborhood method as mentioned at the beginning. Here we use 10 closest users as the neighborhood.
- *N-CDCF-I:* The item-based neighborhood model. We use k=10 closest items as the neighborhood.
- *MF-CDCF:* The MF model described at the beginning. It is run over the concatenated rating matrix that mixes the items of four domains.
- *CMF:* It couples the rating matrices of four domains on the *user* dimension [Singh and Gordon, 2008].

Since this dataset does not explicitly provide memberships of communities, we have to generate communities from data for the BLMA. As the community detection algorithm is not the focus of this paper, we simply employ hierarchical clustering algorithm to generate 50 communities for each domain.

The similarity for the clustering algorithm is computed by:

$$sim^t(u_m, u_n) = \frac{1}{2} s^t(u_m, u_n) + \frac{1}{2|\boldsymbol{A}|} \sum_{d \in \boldsymbol{A}} s^d(u_m, u_n)$$

where $s(u_m, u_n) = (n_{ij}/n_{ij} + 20) cosine\langle u_m, u_n \rangle$, n_{ij} denotes the number of items rated by both u_m and u_n [Koren, 2008], t denotes the target domain, and $\boldsymbol{A}$ denotes other domains. It assigns more weight to the similarity of the target domain t so as to create the domain-specific communities.

The dimensionality of factors and hyper-parameters for all comparative methods are determined by cross validation. The improvement becomes not significant, and even decreases for some models, after the dimensionality is increased above 30.

5.3 Results

We first evaluated the prediction performance using the above constructed training/testing sets for sparse data cases. Table 2 reports the results of all comparative methods.

Table 2: MAE of comparative models for the sparse data cases

	Book		*Music CD*	
Model	75%	20%	75%	20%
MF-SGD	0.597	0.833	0.749	0.942
N-CDCF-U	0.488	0.776	0.701	0.906
N-CDCF-I	0.728	0.850	0.776	1.062
MF-CDCF	0.503	0.753	0.715	0.832
CMF	0.452	0.751	0.686	0.817
BLMA	**0.321**	**0.702**	**0.632**	**0.771**

From Table 2, we can find that the CDCF methods achieve much better performance than the single-domain CF method, i.e., MF-SGD. Therein, our model, the BLMA, significantly outperforms all other comparative methods over all testing cases, and at least 18% improvement is achieved for any case comparing to MF-SGD, which illustrates that the BLMA can better capture the determining factors for predicting ratings even when the user data is not sufficient. The N-CDCF-U also achieves reasonable performance when the data is relative dense, i.e., the 75% training set, but its performance degenerates very rapidly when the data becomes sparser. It is because the neighborhood based method usually fails to find any global similarity among users when the data is sparse [Su and Khoshgoftaar, 2009]. For the MF based CDCF models, the CMF outperforms the MF-CDCF because the CMF provides an effective way to transfer knowledge between domains whereas the MF-CDCF simply aggregates all available data together, ignoring heterogeneities in different domains.

Then, we evaluated the prediction performance for cold-start cases. Figure 4 illustrates the results of all comparative methods. Once again, the BLMA models achieve better performance than other approaches. In this experiment, we find that the CMF lags much behind the BLMA. Although the CMF captures common user factors across all domains, it does not correlate users, nor does it model domain-specific factors. In cold-start cases, the CMF inevitably suffers from the blind-transfer issue. In comparison, the BLMA captures multilevel effects instead of just individual effects. Such design enables the BLMA leveraging the knowledge learned from the domain-level and the community-level data even when the individual-level data are not available. Specially, the BLMA-C (using the community mean cf. § 3.3) outperforms the BLMA-P (using the population mean), which illustrates that leveraging neighbors' feedbacks can more effectively improve the accuracy when a user's data is absent.

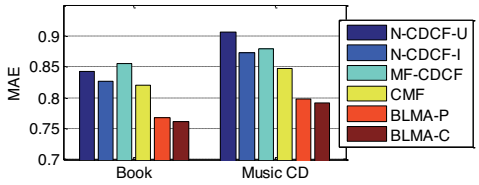

Figure 4: MAE of comparative models for the cold-start cases

Figure 5 (a) plots the sampled domain effects, $\boldsymbol{a}_d$, of all four domains, and the cosine similarities between each pair of $\boldsymbol{a}_d$ are also provided in (b). The larger similarity may reflect the smaller domain heterogeneity to share knowledge. As to a specific effect $a_k \in \boldsymbol{a}_d$, it is positive in some domains, which results in a larger item factor $v_k \in \boldsymbol{v}_d$ contributes more to the rating, i.e. $a_k v_k$. In contrast, a_k is negative in other domains, so the larger v_k (e.g. the price of an item) gives the more negative contribution to the rating, e.g., higher price may yield lower rating. Moreover, the random effects model is often used in the analysis of variance (ANOVA). Figure 5 (c) shows the sampled variances for community random effects, $\boldsymbol{o}_d$, of the *Book* and *Music* domain. A large variance for a community effect, $o_k \in \boldsymbol{o}_d$, reflects users in different communities having quite different views to the item effect $v_k \in \boldsymbol{v}_d$ (e.g. the genre of music) whereas a small variance reflects very close views to v_k over all communities.

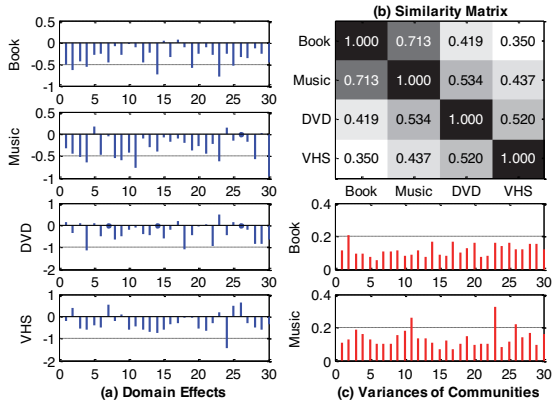

Figure 5: (a) The domain effects of all four domains; (b) The cosine similarity between each pair of domain effects; (c) The variances of community effects of the *Book* and *Music* domain.

6 Conclusions

In this paper, we propose the BLMA approach to solve the CDCF problem by employing a multilevel effects model to overcome the vulnerabilities of flat individual effects model in traditional MF approaches. The experimental results indicate that the BLMA can achieve much better performance than other state-of-the-art models. Since the BLMA provides an integrated approach by coupling the multilevel analysis with the MF, we also expect it to become a general framework for new applications other than the CDCF problem.

Acknowledgement

This work is partially supported by Australian Research Council Discovery grant (DP130102691), Linkage grants (LP120100566 and LP100200774), China National Science Foundation (Granted Number 61073021, 61272438), Research Funds of Science and Technology Commission of Shanghai Municipality (Granted Number 11511500102, 12511502704), Cross Research Fund of Biomedical Engineering of Shanghai Jiaotong University (YG2011MS38).

*Jian Cao is the corresponding author

References

[Berkovsky, *et al.*, 2007] Berkovsky, S., Kuflik, T., Ricci, F.: Cross-Domain Mediation in Collaborative Filtering. In *Proceedings of the 11th international conference on User Modeling*, pp. 355-359. Springer-Verlag (2007)

[Browne, 1998] Browne, W.J.: Applying MCMC methods to multi-level models. *University of Bath* (1998)

[Browne and Draper, 2006] Browne, W.J., Draper, D.: A comparison of Bayesian and likelihood-based methods for fitting multilevel models. *Bayesian Analysis 1*, 473-514 (2006)

[Browne and Draper, 2000] Browne, W.J., Draper, D.: Implementation and performance issues in the Bayesian and likelihood fitting of multilevel models. *Computational statistics 15*, 391-420 (2000)

[Browne, *et al.*, 2001] Browne, W.J., Goldstein, H., Rasbash, J.: Multiple membership multiple classification (MMMC) models. *Statistical Modelling 1*, 103-124 (2001)

[Fortunato, 2010] Fortunato, S.: Community detection in graphs. *Physics Reports 486*, 75-174 (2010)

[Gelman and Hill, 2006] Gelman, A., Hill, J.: Data analysis using regression and multilevel/hierarchical models. *Cambridge University Press* (2006)

[Goldstein, *et al.*, 2007] Goldstein, H., de Leeuw, J., Meijer, E.: Handbook of multilevel analysis. *Springer* (2007)

[Koren, 2008] Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 426-434. ACM (2008)

[Koren, *et al.*, 2009] Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. *Computer 42*, 30-37 (2009)

[Leskovec, *et al.*, 2007] Leskovec, J., Adamic, L.A., Huberman, B.A.: The dynamics of viral marketing. *ACM Trans. Web 1*, 5 (2007)

[Li, 2011] Li, B.: Cross-Domain Collaborative Filtering: A Brief Survey. In *Proceedings of the 2011 IEEE 23rd International Conference on Tools with Artificial Intelligence*, pp. 1085-1086. IEEE Computer Society (2011)

[Li, *et al.*, 2009] Li, B., Yang, Q., Xue, X.: Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction. In *IJCAI*, pp. 2052-2057. Morgan Kaufmann Publishers Inc. (2009)

[Li, *et al.*, 2009] Li, B., Yang, Q., Xue, X.: Transfer learning for collaborative filtering via a rating-matrix generative model. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 617-624. ACM (2009)

[Ma, *et al.*, 2008] Ma, H., Yang, H., Lyu, M.R., King, I.: SoRec: social recommendation using probabilistic matrix factorization. In *Proceeding of the 17th ACM conference on Information and knowledge management*, pp. 931-940. ACM (2008)

[Pan, *et al.*, 2010] Pan, W., Xiang, E.W., Liu, N.N., Yang, Q.: Transfer learning in collaborative filtering for sparsity reduction. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence* (2010)

[Rabe-Hesketh and Skrondal, 2008] Rabe-Hesketh, S., Skrondal, A.: Multilevel and longitudinal modeling using Stata. *STATA press* (2008)

[Raudenbush, 1993] Raudenbush, S.W.: A crossed random effects model for unbalanced data with applications in cross-sectional and longitudinal research. *Journal of Educational and Behavioral Statistics 18*, 321-349 (1993)

[Salakhutdinov and Mnih, 2008] Salakhutdinov, R., Mnih, A.: Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In *Proceedings of the 25th international conference on Machine learning*, pp. 880-887. ACM (2008)

[Singh and Gordon, 2012] Singh, A.P., Gordon, G.: A Bayesian matrix factorization model for relational data. *arXiv preprint arXiv:1203.3517* (2012)

[Singh and Gordon, 2008] Singh, A.P., Gordon, G.J.: Relational learning via collective matrix factorization. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 650-658. ACM (2008)

[Snijders and Bosker, 2011] Snijders, T.A.B.A.B., Bosker, R.J.: Multilevel analysis: An introduction to basic and advanced multilevel modeling. *Sage Publications Limited* (2011)

[Su and Khoshgoftaar, 2009] Su, X., Khoshgoftaar, T.M.: A survey of collaborative filtering techniques. *Adv. in Artif. Intell. 2009*, 2-2 (2009)

[Weston, *et al.*, 2012] Weston, J., Wang, C., Weiss, R., Berenzweig, A.: Latent Collaborative Retrieval. In *Proceedings of the 29th International Conference on Machine Learning*, pp. 9-16. Omnipress (2012)