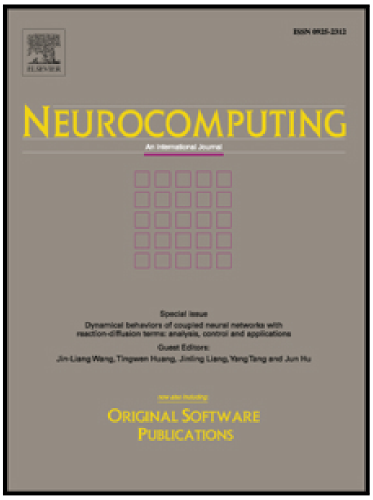
A Probabilistic Framework to Incorporate Mixed-Data Type Features: Matrix Factorization with Multimodal Side Information

Communicated by Steven Hoi


ELSEVIER
ISSN 0925-2312
NEUROCOMPUTING
An International Journal
Special issue
Dynamical behaviors of coupled neural networks with reaction-diffusion terms: analysis, control and applications
Guest Editors:
Jin-Liang Wang, Tingwen Huang, Jinling Liang, Yang Tang and Jun Hu
now also including
ORIGINAL SOFTWARE PUBLICATIONS

A Probabilistic Framework to Incorporate Mixed-Data Type Features: Matrix Factorization with Multimodal Side Information

Mehmet Aktukmak, Yasin Yilmaz, Ismail Uysal

PII: S0925-2312(19)31146-4
DOI: https://doi.org/10.1016/j.neucom.2019.08.019
Reference: NEUCOM 21179

To appear in: *Neurocomputing*

Received date: 4 June 2019
Accepted date: 8 August 2019

Please cite this article as: Mehmet Aktukmak, Yasin Yilmaz, Ismail Uysal, A Probabilistic Framework to Incorporate Mixed-Data Type Features: Matrix Factorization with Multimodal Side Information, *Neurocomputing* (2019), doi: https://doi.org/10.1016/j.neucom.2019.08.019

A Probabilistic Framework to Incorporate Mixed-Data Type Features: Matrix Factorization with Multimodal Side Information

Mehmet Aktukmak, Yasin Yilmaz, Ismail Uysal

University of South Florida, Tampa

Abstract

Recommender systems that exclusively rely on past interactions between the users and the items underperform in settings with very few observations. Their performance at such extreme sparsity can be improved by exploiting the side information of the users and the items including the demographics and the item descriptions. However, such side information is mostly heterogeneous and multimodal, including both numerical and categorical features to yield a non-trivial incorporation process. Researchers have addressed this problem mainly by converting the categorical features into numerical ones or forming numerical similarity matrices. This paper presents a different approach in the form of a novel Bayesian probabilistic generative framework which can effectively incorporate multimodal side information into the matrix factorization (MF-MSI) model. With the help of local quadratic bounds on the categorical likelihoods, we derive a scalable and computationally efficient iterative optimization method based on the variational EM to learn the posterior distributions of latent variables for both the users and the items. A comprehensive experimental study on both simulated and real benchmark datasets demonstrate the proof-of-concept where the additional side information improves both the prediction accuracy and the ranking performance over more than a dozen popular baseline models. Finally, the proposed MF-MSI model claims state-of-the-art performance in the majority of the test scenarios when compared to more recently introduced recommender systems which can also utilize the side information via different techniques.

Keywords: Side information, unsupervised learning, probabilistic generative models, recommender systems, collaborative filtering

1. Introduction

The Internet makes it possible to access an unprecedented amount of consumer content. Naturally, as the number and diversity of the data sources increase, the information retrieval process becomes a challenge of its own. The management of big data has been an active research topic in the machine learning community. For instance, recommender systems personalize content delivery for popular applications such as streaming devices, e-commerce, and online media. Within this context, a successful recommender system can accurately and efficiently guide consumers to the products and information they are looking for.

The ultimate objective of a recommender system is to learn the user patterns explicitly or implicitly by using the available information. Depending on the platform, the information sources can include, i) the past interactions and feed-backs between the users and the items, ii) the demographic information of the users, iii) the features directly related to the items, iv) the social relationships/trust between the users, v) the network structures connecting the users/items based on specific criteria, vi) user-contributed information such as textual reviews and vii) cross-domain knowledge from external domains [1, 2]. Collaborative filtering methods use only the interactions between the users and the items to learn the user patterns. In contrast, content-based methods use only the side information which may be provided by the users (such as gender, age, occupation) or the content provider (such as movie genre, year). The hybrid recommender system combines both approaches for a demonstrably more successful in terms of overall recommendation accuracy for a variety of applications [3]. Specifically, the scenarios where the users and the items have very few observed interactions, called cold-start scenario, represents a particular challenge which can be better resolved by the hybrid recommendation systems that can combine the more readily available and less sparse user and item side information sources with their significantly sparser interactions [1].

Incorporating side information is an intuitive solution for the cold start problem. In most cases, the users and items have available demographic information that can be used as relevant side information. In the case where a user has no interactions (examples include new users or existing users trying out a new category of products) but still some available side information like the age, gender, occupation, etc. the system can still infer a recommendable item for this user based solely on the side information. On the contrary, a warm-start scenario is considered when a user has too many interactions where his/her pattern implies more about the user than their demographic information. A hybrid system should use the information sources efficiently and make an accurate inference for both the cold-start and warm-start settings. However, the process of incorporating side information is non-trivial due to the diversity and heterogeneity of the side information data format. For example, for a movie, the release date corresponds to a numerical feature, whereas its genre has a categorical value. Nonetheless, both should be incorporated appropriately according to the nature of these observations.

Probabilistic generative models are powerful tools that can be used for the datasets, including many missing values. They allow missing data to be handled in a principled way by marginalizing over the distribution of the unobserved variables [4]. Since these models account for the uncertainty of the latent variables, they also handle over-fitting problem, that occurs severely in the case of very sparse data, by regularizing the latent variables with proper priors [5]. Therefore, these models are particularly useful for recommender systems with the condition that one can overcome the challenge of incorporating heterogeneous multimodal side information which represents the main objective of this paper. In this study, we propose a multimodal generative model and a variational Expectation Maximization (EM) algorithm to infer the latent representations of both the users and the items to leverage the emerging variational inference methods [6]. Our method demonstrates the efficient incorporation of mixed data type side information in a scalable probabilistic generative framework. In summary, our contributions can be summarized as follows:

- A novel probabilistic generative model that can incorporate mixed data type side information. The natural parameters of the side information sources are regressed from the latent variables. This allows the incorporation of any data type that can be modeled with the exponential family distribution, although we mainly focus on categorical and numerical features. Since the model is generative, it can make an inference not only in the presence of missing values in the rating matrix but also in the side information as well.

- A fundamental solution to solve the problem of intractable inference, which emerges due to the data type variety of the side information, by deriving a variational EM method that turns the inference into an optimization problem. By using the appropriate local quadratic approximations, the posterior distributions of the user and the item latent variables are approximated as Gaussians motivated by the Bernstein-Von Mises theorem [7].

- A reduced computational complexity which scales with the product of the number of items and the number of users (i.e., only linear in each dimension), which makes it suitable for large datasets.

- The state-of-the-art performance on both synthetic and real datasets when compared to a wide range of well-established baselines as well as some more recent contributions.

The paper is organized as follows. The prior related work is given in Section 2. The model is defined in Section 3.1 and the corresponding inference method is derived accordingly in Section 3.2. Computational complexity analysis is given in Section 3.3. Detailed experimental studies including the simulation results and a wide range of performance comparisons on real-world datasets are presented in Section 4. Lastly, the paper concludes with possible directions for future work in Section 6.

2. Related Work

In recent years, researchers have shown greater interest in incorporating side information specifically to solve the cold-start problem [1, 2]. We group the related studies into four categories based on how the side information is being treated. The category of baseline Matrix Factorization (MF) models consists of algorithms that do not use side information; that is, the only source of information is the rating matrix. Prior based models use the side information to regularize the latent space of the users and the items. Regression-based models incorporate the side information into a common latent space shared by both the interactions and the features. Disjoint models remove the latent space sharing property and assume that the rating and side information are generated independently.

2.1. Baseline MF Models

Linear latent variable models such as principal component analysis (PCA) [5] and matrix factorization (MF) [8] have originally led the way in matrix completion tasks such as recommender systems. In particular, MF has been a milestone in collaborative filtering. In this model, based on the assumption that the sparse rating matrix is low rank, the users and the items are mapped into a joint low dimensional space such that the ratings are modeled as the products of the representations in this space. The model optimizes the latent representations to explain the observed interactions by using Stochastic gradient descent or alternating least squares methods. However, sparse nature of the observed data makes the optimization highly prone to over-fitting. Probabilistic matrix factorization (PMF) [9] extends the MF models by introducing zero-mean Gaussian priors for the latent variables for more robust performance in terms of over-fitting. The priors result in L2 norm regularization for the latent variables if one performs MAP estimation for the model parameters. The regularization strength corresponds to the variances of the Gaussian priors which are optimized via cross-validation procedure. However, it is still prone to over-fitting unless the regularization parameters are chosen carefully. Bayesian probabilistic matrix factorization (BPMF) [10] further extends the PMF model by using Gaussian-Wishart priors for the means and the covariances of the latent variables instead of the standard zero mean and identity covariance. That leads to computing posterior of the latent variables instead of the point estimates, which is useful for modeling uncertainty of the variables. Since the complexity is controlled automatically based on the training data, the model is more robust to the hyper-parameter selection. Some further recent extensions on top of the aforementioned models include, i) the local matrix factorization [11, 12, 13], which extends the PMF model by introducing local estimation emerged from the idea of mixture models [14], ii) the mixture rank approximation, which models the rank of the rating matrix in the mixture model with a Laplacian prior to infer the latent space dimension automatically [15] and iii) neural network models [16, 17, 18, 19] as alternative factorization methods in order to replace the linear models with their non-linear counterparts containing many free parameters, which may cause sensitivity to over-fitting due to the sparse nature of the observations.

2.2. Prior Based Models

Prior based models incorporate the user and item features by forming a prior for the user and item latent vectors. A stochastic process given by a polynomial function of features is used to regularize the latent variables of both users and items in [20]. Applying feature-based regression to the priors of the latent variables instead of zero-mean Gaussian priors (as in PMF) lead the way to incorporate side information where a Monte Carlo EM was used to fit the model. Similarly, in [21], the priors of the user and item latent variables are regressed from the features vectors. Factorized Gaussian priors are given as the regression coefficients with a mean field assumption for variational inference.

Kernelized probabilistic matrix factorization (KMF) [22] model assigns Gaussian process priors to the latent factors. The covariance of the priors is derived from the similarity matrix evaluated from the side information of the users and the items. Recently, in [23], the similarity-like matrix, which is called the user-to-user topic inclusion degree based sparse network, is introduced for social-network link prediction. The network is fused with the observed interaction matrix through a probabilistic model where the side information is used as the mean prior.

2.3. Regression-Based Models

Regression-based models assume that the side information and the latent vectors of the users and the items are linearly dependent. In [24], the BPMF model is extended to incorporate the side information by performing a linear regression on the real-valued features of the users and the items. Dirichlet prior is added to the model for local estimation to improve the performance further where collapsed Gibbs sampling is used to fit the data. In another work [25], probabilistic modeling is combined with matrix factorization where the side information consists of the observed words in the articles. A latent topic space is used for fitting by introducing the latent Dirichlet allocation model [26] with regression in the item latent space for joint estimation. Maximum a posterior (MAP) estimation of the latent variables is performed with the EM algorithm. In [27], the similarity matrices are used within a generative model, i.e. the latent factors are assumed to be the ancestors of the similarity matrices in the graphical model. The regression parameters and the latent factors are optimized jointly. An extended work in [28] introduces locality constraint into the latent space to learn local collective embeddings (LCE). [29] proposes an algorithm where the Gaussian process regression is used to incorporate the real-valued features to the matrix factorization model where the probit likelihood is used for preference ranking. For inference, the EM algorithm is used along with the expectation propagation approximation for non-Gaussian likelihoods. Variational autoencoders are also used in probabilistic learning of feature latent representations [30]. By following [26], the article recommendation is performed by replacing the LDA model with a variational autoencoder. Additionally, especially for contextual recommendation, factorization machines [31] and tensor factorization methods [32], that can use additional information beyond rating matrix, are proposed. Recently, several algorithms [33, 34, 35] are developed based on tensor factorization to transfer knowledge from other domains as side information to alleviate the cold-start problem.

2.4. Disjoint Models

Contrary to the previously discussed methods, there is a work in the literature that the rating matrix and side information are assumed to be generated independently, i.e. they don't share the same latent space. In [36], matrix factorization is augmented with regression against the real-valued side information by using a weighted scheme. The side information is assumed to be marginally independent. In [37], the normalized features are added to the latent vectors and stochastic gradient descent is performed as in MF. In [38], side information is used to compute multiple item similarity functions. These functions are weighted for each user with trained weights to make personalized recommendations. In [39], a dense submatrix is extracted from the rating matrix by selecting the users and items with large numbers of interactions. Matrix factorization is then performed to find latent factors of the corresponding users and items. Afterwards, a linear regression model is employed to relate the latent factors and the similarity matrices of the users and the items where the regression weights are evaluated with the selected user/item latent factors. The resulting algorithm is called DecRec. A similar method proposed in [40] uses concatenated attributes instead of a similarity matrix. For warm-start users and items, latent vectors are evaluated by using the factor model to learn a mapping between the attributes and latent vectors. In [41], independency assumption holds for social and geographical information for the task of a point of interest estimation to fuse the sources of information via matrix factorization. A neural network model proposed in [18], in which concatenated item and user features are fed to two auto-encoders to learn low dimensional representations. These representations are then added to the MF model whose networks and latent vectors are jointly optimized.

The proposed algorithm falls in the general category of regression-based models. The differences between our algorithm and the aforementioned models are several folds. First, all the models treat the side information as uni-modal, i.e. the side information is assumed to be of a single data type. Majority of the models [27, 28, 38, 39, 40, 22] handle the mixed data problem by pre-processing the similarity measures to form a real-valued similarity matrix as the side information. However, that creates a significant burden on computational costs and memory requirements to the extent where these approaches become unscalable for very large datasets as discussed in Section 4. Another problem is their performance relies heavily on the selection criteria of similarity metric which is not straightforward to compute for mixed-data type features. In contrast, the proposed model does not require any pre-processing, i.e the feature dimension is preserved which allows the selection of a lower dimensional latent space for a demonstrably better trade-off between performance and scalability. The mixed data problem is solved by a principle probabilistic generative approach where the real-valued, categorical and binary features are modeled by using Gaussian, categorical and Bernoulli distributions respectively. To demonstrate the improvements in performance, we pick and compare the representative recent algorithms from all four categories. The comparison details are presented in Section 4.

3. Proposed Model

3.1. Model Definition

In this section, we describe the details of the proposed model. The model is developed with multimodal side information including one multivariate real-valued and one categorical observation for both the users and the items, and with the sparse rating matrix formed by the interactions. The ultimate goal is to

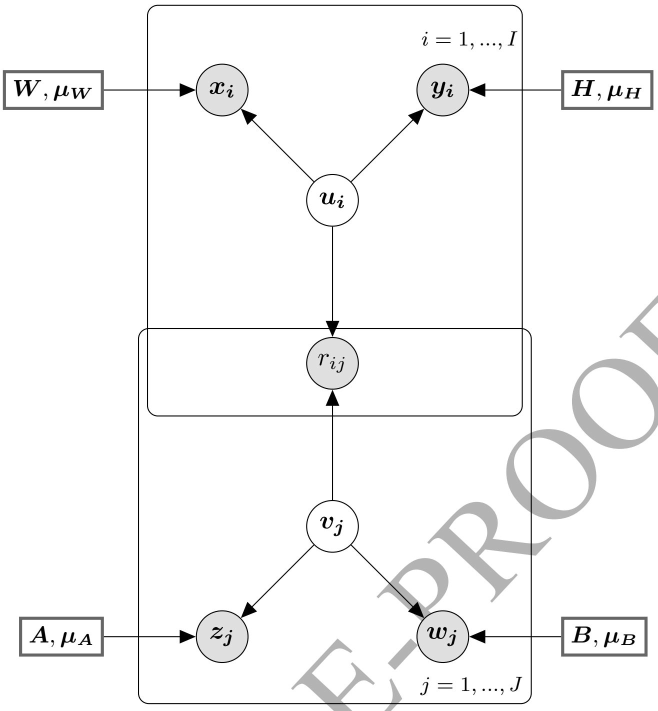


Figure 1: Graphical model representation of the proposed MF-MSI method. The upper plate is for the users, the lower plate is for the items, and the intersection is for the ratings. $\boldsymbol{x}_i$ and $\boldsymbol{z}_j$ denote real-valued side information while $\boldsymbol{y}_i$ and $\boldsymbol{w}_j$ denote categorical side information.

infer the posterior distributions of the user and the item latent variables to explain both the observed ratings and associated side information. The graphical representation of the proposed model is shown in Figure 1. In the probabilistic model, $\boldsymbol{u}_i \in R^K$ corresponds to the latent variable associated with user i and $\boldsymbol{v}_j \in R^K$ corresponds to the latent variable associated with item j. Zero-mean spherical Gaussian priors are assumed for these multivariate latent variables as follows:

$$p(\boldsymbol{u}_i) = \mathcal{N}(\boldsymbol{u}_i | \boldsymbol{0}_K, \lambda_u^{-1} \boldsymbol{I}_K), \tag{1}$$

$$p(\boldsymbol{v}_j) = \mathcal{N}(\boldsymbol{v}_j | \boldsymbol{0}_K, \lambda_v^{-1} \boldsymbol{I}_K), \tag{2}$$

where λ_u and λ_v are the precision hyperparameters for the distributions. K is the latent space dimension. Instead of the zero-mean prior, it is trivial to use Gaussian-Wishart priors as in [10] for fully Bayesian treatment to prevent over-fitting that can easily occur in the case that the regularization precision parameters are not tuned correctly within a validation set. However, for simplicity of derivations, we stick to the zero-mean spherical priors. The generative process assumes that both the real-valued and the categorical side information represented by $\boldsymbol{x}_i \in R^{D_u}$ and $\boldsymbol{y}_i \in R^{M_u}$, respectively, are generated from $\boldsymbol{u}_i$ through the model parameters via regression. We hold on to this assumption since many regression-based models in the literature [24, 25, 42, 28] have proved that modeling the generation process for the features linearly through the natural parameters of their distributions is a valid assumption and results in reasonable performance. Accordingly, the conditional probability of $\boldsymbol{x}_i$ is given as a Gaussian distribution:

$$p(\boldsymbol{x}_i|\boldsymbol{u}_i) = \mathcal{N}(\boldsymbol{x}_i|\boldsymbol{W}\boldsymbol{u}_i + \boldsymbol{\mu}_{\boldsymbol{W}}, \boldsymbol{\Sigma}_{\boldsymbol{x}}), \tag{3}$$

where $\boldsymbol{W} \in R^{D_u \times K}$, $\boldsymbol{\mu}_{\boldsymbol{W}} \in R^{D_u}$ and $\boldsymbol{\Sigma}_{\boldsymbol{x}} \in R^{D_u \times D_u}$ are the model parameters associated with the real-valued user side information. For $\boldsymbol{y_i}$, the categorical conditional distribution is assigned as follows:

$$p(\boldsymbol{y}_i|\boldsymbol{u}_i) = Cat(\boldsymbol{y}_i|\mathcal{S}(\boldsymbol{H}\boldsymbol{u}_i + \boldsymbol{\mu}_{\boldsymbol{H}})), \tag{4}$$

where $\boldsymbol{H} \in R^{M_u \times K}$ and $\boldsymbol{\mu}_{\boldsymbol{H}} \in R^{M_u}$ are the model parameters that are associated with the categorical side information. $\mathcal{S}$ is the Softmax function that maps the natural parameters of the categorical distribution to the probability of each class. The natural parameters of both distributions are linearly modeled, i.e., $\boldsymbol{\eta}_{G,i} = \boldsymbol{W}\boldsymbol{u}_i + \boldsymbol{\mu}_{\boldsymbol{W}}$ for the Gaussian distribution, and $\boldsymbol{\eta}_{C,i} = \boldsymbol{H}\boldsymbol{u}_i + \boldsymbol{\mu}_{\boldsymbol{H}}$ for the categorical distribution. $\boldsymbol{\eta}_{G,i}$ corresponds to the mean of the Gaussian distribution, and $\boldsymbol{\eta}_{C,i} = [\log \frac{p_1}{p_{M_u+1}}, \ldots, \log \frac{p_{M_u}}{p_{M_u+1}}]^T$ where $\{p_1, \ldots, p_{M_u+1}\}$ are the probabilities in the categorical distribution. A symmetric configuration is used for the item side with the following distributions:

$$p(\boldsymbol{z}_j|\boldsymbol{v}_j) = \mathcal{N}(\boldsymbol{z}_j|\boldsymbol{A}\boldsymbol{v}_j + \boldsymbol{\mu_A}, \boldsymbol{\Sigma_z}), \tag{5}$$

$$p(\boldsymbol{w}_j|\boldsymbol{v}_j) = Cat(\boldsymbol{w}_j|\mathcal{S}(\boldsymbol{B}\boldsymbol{v}_j + \boldsymbol{\mu_B})), \tag{6}$$

where $\boldsymbol{z}_j \in R^{D_v}$ and $\boldsymbol{w}_j \in R^{M_v}$ represent the real valued and the categorical side information for item j, respectively. The corresponding model parameters are $\boldsymbol{A} \in R^{D_v \times K}$, $\boldsymbol{\mu_A} \in R^{D_v}$, $\boldsymbol{\Sigma_z} \in R^{D_v \times D_v}$, $\boldsymbol{B} \in R^{M_v \times K}$ and $\boldsymbol{\mu_B} \in R^{M_v}$. Finally, the rating matrix is assumed to be generated with the interactions between the user and the item latent variables. The conditional probability for each rating is modeled with the precision parameter c as follows:

$$p(r_{ij}|\boldsymbol{u}_i, \boldsymbol{v}_j) = \mathcal{N}(r_{ij}|\boldsymbol{u}_i^T\boldsymbol{v}_j, c^{-1}). \tag{7}$$

3.2. Inference

Next, we infer the posterior distributions of the user and the item latent variables $\boldsymbol{u}_i$ and $\boldsymbol{v}_j$ given the observed ratings and the multimodal side information. We also find the Maximum Likelihood estimations of the global model parameters that are collected in the set Θ as

$$\Theta = \{\boldsymbol{W}, \boldsymbol{\mu_W}, \boldsymbol{H}, \boldsymbol{\mu_H}, \boldsymbol{A}, \boldsymbol{\mu_A}, \boldsymbol{B}, \boldsymbol{\mu_B}, \boldsymbol{\Sigma_x}, \boldsymbol{\Sigma_z}, c\}.$$

The model has two hyper-parameters that is included in the set $\zeta = \{\lambda_u, \lambda_v\}$. In order to fit the latent variable models, the EM algorithm, which maximizes a lower bound for the marginal likelihood, provides a powerful solution [4]. However, an exact EM algorithm cannot be used to infer the model parameters due to the intractable posteriors of the latent variables for the categorical likelihoods. Specifically, the complete data likelihood is given for user i and item j by following generative process as follows:

$$L_{ij} = p(\boldsymbol{u}_i)p(\boldsymbol{x}_i|\boldsymbol{u}_i)p(\boldsymbol{y}_i|\boldsymbol{u}_i)p(\boldsymbol{v}_j)p(\boldsymbol{z}_j|\boldsymbol{v}_j)p(\boldsymbol{w}_j|\boldsymbol{v}_j)p(r_{ij}|\boldsymbol{u}_i, \boldsymbol{v}_j). \tag{8}$$

The likelihood consists of the categorical likelihoods of $p(\boldsymbol{y}_i|\boldsymbol{u}_i)$ and $p(\boldsymbol{w}_j|\boldsymbol{v}_j)$ which make an exact inference intractable. Instead, motivated by the Bernstein-von Mises theorem [7], we use variational inference by restricting the posterior distributions only to Gaussians to make the lower bound tractable. The variational EM approach is used by defining the local quadratic bounds for categorical likelihoods where Bohning bound has been shown to provide a useful lower bound [43, 44, 45]. This bound is obtained by locally approximating the log-sum-exp (lse) function for the log-likelihood of multinomial and categorical distributions [46]. The approximation is performed around a point called the free variational parameter. The log likelihood of the categorical distribution of user side information after applying Bohning bound can be written as follows:

$$\begin{aligned}\log p(\boldsymbol{y}_i|\boldsymbol{u}_i) &= \log \frac{e^{\boldsymbol{y}_i^T \boldsymbol{\eta}_{C,i}}}{1+\sum_{k=1}^{M_u} e^{\boldsymbol{\eta}_{C,i,k}}} \\ &= \boldsymbol{y}_i^T \boldsymbol{\eta}_{C,i} - \mathrm{lse}(\boldsymbol{\eta}_{C,i})) \\ &\geq \boldsymbol{y}_i^T \boldsymbol{\eta}_{C,i} - \frac{1}{2}\boldsymbol{\eta}_{C,i}^T \boldsymbol{F}_{\boldsymbol{u}} \boldsymbol{\eta}_{C,i} + \boldsymbol{g}_i^T \boldsymbol{\eta}_{C,i} - \boldsymbol{e}_i \\ &\geq \boldsymbol{y}_i^T(\boldsymbol{H}\boldsymbol{u}_i + \boldsymbol{\mu}_{\boldsymbol{H}}) - \frac{1}{2}(\boldsymbol{H}\boldsymbol{u}_i + \boldsymbol{\mu}_{\boldsymbol{H}})^T \boldsymbol{F}_{\boldsymbol{u}}(\boldsymbol{H}\boldsymbol{u}_i + \boldsymbol{\mu}_{\boldsymbol{H}}) \\ &\quad + \boldsymbol{g}_i^T(\boldsymbol{H}\boldsymbol{u}_i + \boldsymbol{\mu}_{\boldsymbol{H}}) - \boldsymbol{e}_i,\end{aligned} \tag{9}$$

where $\boldsymbol{\eta}_{C,i,k}$ are the elements of the vector $\boldsymbol{\eta}_{C,i}$, and the lse function is given by $\mathrm{lse}(\boldsymbol{\eta}_{C,i}) = \log(1+\sum_{k=1}^{M_u} e^{\boldsymbol{\eta}_{C,i,k}})$. When the quadratic bound approximation is used, the lower bound to the complete data log-likelihood also becomes quadratic which lets the posteriors be approximated as Gaussian distributions. This is a reasonable approximation for large number of features $(D_u + M_u + I)$ since the conditions of Bernstein-von Mises theorem are satisfied under the exponential family models with a Gaussian prior [7]. The intermediate parameters that are used within the bound are given as follows [43]:

$$\boldsymbol{F}_{\boldsymbol{u}} = \frac{1}{2}(\boldsymbol{I}_{M_u} - \frac{1}{M_u+1}\mathbf{1}_{M_u}\mathbf{1}_{M_u}^T), \tag{10}$$

$$\boldsymbol{g}_i = \boldsymbol{F}_{\boldsymbol{u}}\boldsymbol{\psi}_i - \mathcal{S}(\boldsymbol{\psi}_i), \tag{11}$$

$$\boldsymbol{e}_i = \frac{1}{2}\boldsymbol{\psi}_i^T \boldsymbol{F_u} \boldsymbol{\psi}_i - \mathcal{S}(\boldsymbol{\psi}_i)^T \boldsymbol{\psi}_i + \mathrm{lse}(\boldsymbol{\psi}_i), \tag{12}$$

where $\boldsymbol{\psi}_i$ is the free variational parameter around which the lse function is approximated. At each iteration, this parameter is updated as well to change the local approximation point. Gaussian log-likelihoods for user i that appears in the complete data log-likelihood are given as follows:

$$\log p(\boldsymbol{u}_i) = -\frac{K}{2}\log(2\pi) - \frac{1}{2}\log|\lambda_u^{-1}\boldsymbol{I}_K| - \frac{\lambda_u}{2}\boldsymbol{u}_i^T \boldsymbol{u}_i, \tag{13}$$

$$\log p(r_{ij}|\boldsymbol{u}_i, \boldsymbol{v}_j) = -\frac{1}{2}\log(2\pi) - \frac{1}{2}\log c - \frac{c}{2}(r_{ij} - \boldsymbol{u}_i^T \boldsymbol{v}_j), \tag{14}$$

$$\log p(\boldsymbol{x}_i|\boldsymbol{u}_i) = -\frac{D_u}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{\Sigma_x}| - \frac{1}{2}(\boldsymbol{x}_i - \boldsymbol{W}\boldsymbol{u}_i - \boldsymbol{\mu_W})\boldsymbol{\Sigma_x}^{-1}(\boldsymbol{x}_i - \boldsymbol{W}\boldsymbol{u}_i - \boldsymbol{\mu_W}). \tag{15}$$

The log-likelihoods for the items are similar due to the symmetry of the model and will not be replicated to avoid clutter. The lower bound for the complete data log-likelihood is found by the summation of the log-likelihoods of each factor in Eq.8. In the EM algorithm, taking the expectation of this bound with respect to the posterior distributions of the latent variables $\boldsymbol{u}_i$ and $\boldsymbol{v}_j$ first by using the old model parameter values and later by maximizing this expectation with respect to these parameters will yield a new parameter set [4].

E-Step: Specifically, we first obtain the means and the variances of the Gaussian approximation for the posteriors of $\boldsymbol{u}_i$ and $\boldsymbol{v}_j$. These can be derived by completing the square by collecting quadratic and linear terms in the log-likelihood to form Gaussian likelihoods [47]. It is important to note that since a lower bound is used, the variational posterior distributions are obtained as $q(\boldsymbol{u}_i) = \mathcal{N}(\boldsymbol{u}_i|\boldsymbol{m}_{\boldsymbol{u}i}, \boldsymbol{\Sigma}_{\boldsymbol{u}i})$ for $\boldsymbol{u}_i$ and $q(\boldsymbol{v}_j) = \mathcal{N}(\boldsymbol{v}_j|\boldsymbol{m}_{\boldsymbol{v}j}, \boldsymbol{\Sigma}_{\boldsymbol{v}j})$ for $\boldsymbol{v}_j$ instead of the exact posteriors. The E-step equations for the variational parameters $\boldsymbol{m}_{\boldsymbol{u}i}$ and $\boldsymbol{\Sigma}_{\boldsymbol{u}i}$ of $q(\boldsymbol{u}_i)$ are given as follows:

$$\boldsymbol{\Sigma}_{\boldsymbol{u}i} = (\lambda_u \boldsymbol{I}_K + \boldsymbol{H}^T \boldsymbol{F_u} \boldsymbol{H} + \boldsymbol{W}^T \boldsymbol{\Sigma_x}^{-1} \boldsymbol{W} + c(E[\boldsymbol{V}\boldsymbol{O}_i \boldsymbol{V}^T]))^{-1}, \tag{16}$$

$$\boldsymbol{m}_{\boldsymbol{u}i} = E[\boldsymbol{u}_i] = \boldsymbol{\Sigma}_{\boldsymbol{u}i}(c(E[\boldsymbol{V}]\boldsymbol{O}_i\boldsymbol{r}_{\boldsymbol{i}}) + \boldsymbol{H}^T(\boldsymbol{y}_i + \boldsymbol{g}_i - \boldsymbol{F}_{\boldsymbol{u}}\boldsymbol{\mu}_{\boldsymbol{H}}) + \boldsymbol{W}^T\boldsymbol{\Sigma}_{\boldsymbol{x}}^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_{\boldsymbol{W}})), \tag{17}$$

$$E[\boldsymbol{u}_i\boldsymbol{u}_i^T] = \boldsymbol{\Sigma}_{\boldsymbol{u}i} + E[\boldsymbol{u}_i]E[\boldsymbol{u}_i^T], \tag{18}$$

where $\boldsymbol{V} = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_J]$ and $\boldsymbol{r}_i = [r_{i1}, \ldots, r_{iJ}]^T$. $\boldsymbol{O}_i$ is a $J \times J$ diagonal matrix whose entries are the binary indicators of the observed ratings of each item for user i to calculate the sufficient statistic by summing only the second moments of the items rated by user i. c is a global parameter that weighs this statistic to maintain a balance between the ratings and the side information. The first term in Eq.16 is the prior precision given for $\boldsymbol{u}_i$ that prevents over-fitting. The second and the third terms correspond to the contributions of the categorical and real-valued side information respectively. Note the fact that the second term depends only on the global parameters $\boldsymbol{H}$ and $\boldsymbol{F}_{\boldsymbol{u}}$ instead of the instances. The last term couples the user posterior covariance with the second moment of the items through a coefficient c. The optimal parameter c is estimated during the M-step by taking the sparsity of the dataset into account. After calculating the posterior covariance, the posterior mean is calculated by using Eq.17. The first term in this equation couples the mean of the item latent variables with the observed ratings. $\boldsymbol{O}_i$ term effectively includes only those item variables that are rated by the user i. c is used to weigh this coupling term with respect to the second and third terms corresponding to the observations with the categorical and the Gaussian side information respectively. The sum of these terms is multiplied with the posterior covariance to calculate the posterior mean of the latent variable of the user i. Next, the second moment of each user is calculated since the coupling between the items and the users in the posterior covariance calculation for item latent variable $\boldsymbol{v}_j$ appears through this statistic, as shown in Eq.16 for the item second moment. Finally, the variational free parameters for each user and item are updated in the E-step for appropriate local approximation. Optimal update for a user which has been shown in [46, 44] is given as follows

$$\boldsymbol{\psi}_i = \boldsymbol{H}E[\boldsymbol{u}_i] + \boldsymbol{\mu}_{\boldsymbol{H}}. \tag{19}$$

The equations for the item vector $\boldsymbol{v}_j$ are similar (due to the symmetrical graphical model) with the same form but different parameters. The sum of second moments over all items and users is used as a sufficient statistic to evaluate the M-step.

M-Step: By using the predicted posterior distributions of $\boldsymbol{u}_i$ and $\boldsymbol{v}_j$ after each E-step, the model parameters are estimated point-wise to maximize the lower bound of the expected complete-data log-likelihood, which corresponds to the M-step of the EM algorithm. To find the update equations for the model parameters, the derivative of the expectation of complete-data log-likelihood with respect to each model parameter is evaluated. For the Gaussian modality of the user side, the update equations for the global parameters $\boldsymbol{W}$, $\boldsymbol{\mu}_{\boldsymbol{W}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{x}}$ are obtained as follows.

$$\boldsymbol{W} = \Big[\sum_i (\boldsymbol{x}_i - \boldsymbol{\mu}_{\boldsymbol{W}})E[\boldsymbol{u}_i]^T\Big]\Big[\sum_i E[\boldsymbol{u}_i\boldsymbol{u}_i^T]\Big]^{-1}, \tag{20}$$

$$\boldsymbol{\mu}_{\boldsymbol{W}} = \frac{1}{I}\sum_i \boldsymbol{x}_i, \tag{21}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{x}} = \mathrm{diag}\Big\{\frac{1}{I}\sum_i (\boldsymbol{x}_i - \boldsymbol{\mu}_{\boldsymbol{W}})(\boldsymbol{x}_i - \boldsymbol{\mu}_{\boldsymbol{W}})^T - (\boldsymbol{x}_i - \boldsymbol{\mu}_{\boldsymbol{W}})E[\boldsymbol{u}_i]^T\boldsymbol{W}^T\Big\}. \tag{22}$$

These are exactly the same update equations as in the factor analysis models [14]. For the categorical modality, following the same approach, the update equations are obtained for the global parameters $\boldsymbol{H}$ and $\boldsymbol{\mu}_{\boldsymbol{H}}$ as follows:

$$\boldsymbol{H} = \Big[\sum_i (\boldsymbol{F}_{\boldsymbol{u}}^{-1}(\boldsymbol{y}_i + \boldsymbol{g}_i) - \boldsymbol{\mu}_{\boldsymbol{H}})E[\boldsymbol{u}_i]^T\Big]\Big[\sum_i E[\boldsymbol{u}_i\boldsymbol{u}_i^T]\Big]^{-1}, \tag{23}$$

$$\boldsymbol{\mu}_{\boldsymbol{H}} = \frac{1}{I}\sum_i \left\{\boldsymbol{F}_{\boldsymbol{u}}^{-1}(\boldsymbol{y}_i + \boldsymbol{g}_i) - \boldsymbol{H}E[\boldsymbol{u}_i]\right\}. \tag{24}$$

Lastly, the precision parameter c is updated in the same way as follows,

$$c = \frac{1}{|\Omega|} \sum_{i,j \in \Omega} (r_{ij} - \boldsymbol{u}_i^T \boldsymbol{v}_j)^2, \tag{25}$$

where Ω is the set of index pairs $\{i, j\}$ of the observed ratings and $|\Omega|$ is the cardinality of the set. The successive E and M-steps are performed until all the parameters converge to steady-state values.

Once the model is trained and the posterior distributions for all users and items are obtained, a user's score on an item is predicted for the purpose of making a personalized recommendation. A straightforward approach is to compute the inner products of the user and the item posterior means for the unobserved ratings as follows:

$$\hat{r}_{ij} = E[\boldsymbol{u}_i]^T E[\boldsymbol{v}_j] = \boldsymbol{m}_{\boldsymbol{u}\boldsymbol{i}}^T \boldsymbol{m}_{\boldsymbol{v}j}. \tag{26}$$

3.3. Computational Complexity

In this section, we analyze each step of the algorithm in terms of their computational complexities to develop a general understanding of the scalability of the proposed approach. We start with the covariance computation in the E-step, given by Eq.16. The Gaussian modality term inside the summation requires multiplication of a $K \times D_u$ matrix with its transpose which results in $O(K^2 D_u)$. Note that multiplication with the diagonal term $\boldsymbol{\Sigma}_{\boldsymbol{x}}^{-1}$ in Eq.16 has no additional cost. Similarly, the categorical modality term has a complexity of $O(K^2 M_u)$, where multiplication with $\boldsymbol{F}_{\boldsymbol{u}}$ does not increase the complexity due to its special form given in Eq.10. The last term which couples the item second moments with the user posterior covariance requires multiplication of a $K \times J$ matrix with its transpose, resulting in $O(JK^2)$ computations. Hence, for each user, the overall computation of the posterior covariance matrix in Eq.16 requires $O(K^2(M_u + D_u + J + K))$. We can assume that for a typical recommender system J, (the number of items), is very large compared to Du and M_u. Moreover, the number of latent dimensions K is also typically chosen much smaller than I and J, hence the complexity reduces to $O(JK^2)$ for Eq.16.

The computation of the posterior mean in Eq.17 includes only matrix-vector multiplications that require $O(KJ)$ computations. The computation of second moment in Eq.18 requires $O(K^2)$ computations. Finally, for all users, we repeat the computations in Eq.16-Eq.18, resulting in $O(IJK^2)$ asymptotic complexity. Due to the symmetry of the proposed model for categorical posterior, we similarly have a complexity of $O(IJK^2)$, thus the total cost of the E-step is $O(IJK^2)$.

The computation of $\boldsymbol{W}$ in the M-step requires multiplication of two terms. The first one requires a summation over multiplication of $D_u \times 1$ and $1 \times K$ vectors which result in $O(IKD_u)$. The inversion in the second term requires $O(K^3)$ and the multiplication of the two terms requires $O(D_u K^2)$. Subsequently, the total cost is $O(IKD_u)$. Similarly, the computation of $\boldsymbol{\Sigma_x}$ has $O(IKD_u)$ complexity, and for $\boldsymbol{H}$ the complexity is $O(IKM_u)$. Their joint complexity is $O(IK(D_u + M_u))$. Due to the symmetry, the item side parameters need $O(JK(D_v + M_v))$ computations, which makes the total cost of a single M step as $O(IK(D_u + M_u) + JK(D_v + M_v))$.

In terms of the overall computational complexity of a single EM iteration, the focus is on the number of users I and the number of items J as they will significantly outweigh any other system parameter in a typical large scale implementation. We see that while the complexity of the E-step scales by IJ, the M-step scales by $(I + J)$. Conclusively, one can say that for large I and J, the E-step will dominate the computational load with an overall model scaling factor of IJ. Since the complexity linearly scales with both the number of users and the number of items, the proposed algorithm is competitively scalable for big data applications in terms of time computational complexity.

4. Experimental Results

4.1. Evaluation Models

In this section, we briefly describe the models which are included for the purpose of comprehensive performance evaluation. For labeling purposes, the proposed model will be called the MF-MSI method [1]. The other models can be categorized into three groups. The first group includes some baseline MF algorithms that are related to the proposed algorithm [2]. The second group consists of extensively used standard benchmark algorithms. The last group consists of the recent algorithms that can incorporate side information, which are selected from the categories defined in Section 2.

4.1.1. Baseline MF Algorithms

BPMF: In this model, the side information is not incorporated. The user and item latent variables are inferred by using only the sparse rating matrix. In this baseline model, the second and third terms in both the summations of both Eq.16 for posterior covariance and Eq.17 for mean estimation are simply removed. Since there are no categorical likelihoods in this altered complete data likelihood, the exact EM solution is applied instead of the variational EM. This particular baseline model resembles the well-known approaches in the literature such as Bayesian Probabilistic Matrix Factorization (BPMF)[10] and Factor Analysis [14]. In fact, the model definition here is the same as BPMF but the inference is performed via EM instead of Markov Chain Monte Carlo (MCMC), hence, this algorithm will be labeled as BPMF in the comparative studies.

PMF: A gradient-based optimization is used to find the point estimates of the parameters and the MAP estimates of the latent variables which maximize the complete data log-likelihood [9]. This method does not use the posterior covariances of latent variables which makes it more sensitive to the hyper-parameters [43]. PMF method is implemented by removing the side information terms in Eq.16.

4.1.2. Standard Algorithms

NormalPredictor: The unknown ratings are predicted by sampling from a normal distribution whose mean and variances are estimated from the training

[1] The code is available at `https://github.com/maktukmak/MF-MSI`.

[2] Surprise package is used for the implementation of some algorithms in this category.

data using the Maximum Likelihood Estimation.

BaselineOnly: The global, the user and the item-specific means are evaluated by using the ratings in the training dataset. Predictions are performed by summing up the mean values [48].

KNN Models: In the user-based neighborhood models, the similarity between users is computed by using the cosine distance and the predictions are evaluated by linearly weighting the predictions of the k-neighbors with the pre-computed similarity values. We considered four different types of KNN models. KNNBasic is performed with raw ratings. KNNWithMeans takes into account the mean of the user and the item ratings. KNNWithZScore incorporates the z-score normalization for each user. KNNBaseline uses the baseline ratings computed by summing up the global, the user and the item mean rating [49].

SVD: This model factorizes the rating matrix into two low dimensional user and item matrices. The global mean, the user bias and the item bias are incorporated in the model. Stochastic gradient descent is used to optimize low dimensional matrices and biases. It is similar to the PMF model but also includes biases [8].

NMF: This model is similar to SVD but the latent factors are forced to be positive. It is recommended for use particularly in datasets with only positive interactions [50, 51].

SVDpp: This is an extension of SVD taking into account implicit feedback. A new set of item factors is introduced to capture implicit ratings. All the factors are optimized along with biases by using a gradient-based approach [49, 52].

SlopeOne: Average differences between the ratings of the target item and the items rated by the other users are evaluated in a pairwise manner [53].

CoClustering: K-means algorithm is used to form the clusters for the users and the items and the co-cluster for the ratings. The means of these three clusters are summed up to find the prediction of an unknown rating [54].

4.1.3. Recent State-of-the-art Algorithms

LCE: A matrix factorization model which can incorporate side information via collective factorization. This model is similar to the proposed MF-MSI in terms of the common latent space utilization. However, the model can only incorporate real-valued side information. For mixed data types, the side information is preprocessed to form real-valued similarity matrices by using RBF kernels [28].

DecRec: This model extracts a submatrix from the rating matrix, which is dense enough to be completed with low error rates. The completion is performed via classical Matrix Factorization by evaluating the latent factors. The cosine similarity matrices for the users and the items are computed by using the side information. Linear regression is performed to find the latent factors of the users and the items which are excluded by the submatrix by using similarity matrices and in-submatrix latent factors [39].

KMF: Kernelized matrix factorization uses Gaussian process priors to regulate the columns of the latent matrices as opposed to the rows as in classical PMF models. The covariance matrices of the priors are simply assigned as similarity matrices formed by using the side information of the users and the items [22].

LightFM: This model incorporates mixed data type side information by adding the features from the metadata of the users and the items to the classical matrix factorization model. This model is not probabilistic and gradient-based optimization is used to find the point estimates of the latent vectors [37].

4.2. Datasets

Four different datasets are used in the experimental study. The first one is a synthetic dataset generated, i) to analyze the model behavior with respect to varying levels of sparsity, ii) to assess the convergence property and iii) to observe the computational load and scalability with respect to the growing dataset. The comparative study includes three different variations of the MovieLens dataset with the different number of interactions (100K, 1M, and 10M) commonly used as the official benchmark datasets in recommender system applications to assess small to large scale performance. A specific advantage of MovieLens dataset is the availability of multimodal side information for both the users and the items. The statistics of each dataset are summarized in Table 1.

Synthetic: The synthetic dataset is generated by following the generative process described in section 3.1. The rating matrix is generated by randomly removing a fraction of the generated values. In addition to the sparse rating matrix, the fully observed multimodal side information is generated for both the users and the items. The Gaussian modality dimensions are chosen as $D_u = 3$ and $D_v = 3$. Two categorical modalities are incorporated for each side such that $M_u = [5\ 3]$ and $M_v = [5\ 3]$ which means that the first and the second categorical features have 6 and 4 classes (where the last class used as the pivot) respectively. The dimension of the latent space is fixed as $K = 3$. λ_u, λ_v and c are fixed as 1. The number of the users/items (I, J) and the missing value fractions are varied according to the specifications of the experiments.

MovieLens 100K: MovieLens 100K is one of the most popular small-scale recommender system datasets used for benchmarking. The rating matrix is generated by the interactions of 943 users with 1682 items. The number of interactions is 100K which makes the fraction of missing values 0.937. The dataset has fully observed user side information such as age, gender, occupation and zip code. We model the age information as a univariate Gaussian modality and gender and occupation information as categorical modalities with 2 and 21 classes, respectively. For the item side, the dataset has movie title, release date, and genre. We model the release date as a univariate Gaussian. Genre is a 21-dimensional non-1-of-K binary indicator. Hence, the genre information is modeled as 21 different categorical modalities where each one can have two values (Bernoulli distribution).

MovieLens 1M: Similarly, MovieLens 1M is generated by approximately one million interactions of 6040 users on 3883 items which makes the sparsity around 0.957. It has the same metadata as MovieLens 100K dataset for both the users and the items.

MovieLens 10M: Finally, MovieLens 10M is a large-scale dataset that provides approximately 10M interactions of 71567 users on 10681 items. The fraction of missing values, in this case, is 0.987. Unlike the previous two datasets, the user side information is not officially provided. For the item side informa-

Table 1: MovieLens Dataset Statistics.

Statistics	MovieLens-100K	MovieLens-1M	MovieLens-10M
# Users	943	6040	71567
# Items	1682	3883	10681
# Ratings	100000	1000209	10000054
Rating Sparsity	0.937	0.957	0.987
Real-valued info for users	Age	Age	-
Categorical info for users	Gender, occupation	Gender, occupation	-
Real-valued info for items	Release	Release	Release
Categorical info for items	Genre	Genre	Genre

tion, we similarly have the genre and release date like the previous two datasets.

4.3. Evaluation Metrics

To compare the performance of the algorithms described in Section 4.1, two performance evaluation metrics are used throughout the study. Mean Square Error(MSE) is used to assess the performance when explicit rating prediction is performed, which is evaluated as follows:

$$MSE = \frac{1}{|\Omega_{test}|} \sum_{i,j \in \Omega_{test}} (\hat{r}_{ij} - r_{ij})^2$$

where Ω_{test} is the set in which the indices of the test interactions are stored.

The second metric is "recall" that measures the ranking performance of the model. Generally, recall is a more practical assessment of a recommender system as it directly measures the recommendation performance. In the special case of movie datasets, the evaluation is given as follows:

$$Recall@L = \frac{number\ of\ movies\ the\ user\ liked\ in\ the\ top\ L\ recommendations}{total\ number\ of\ movies\ the\ user\ liked}$$

where L is the number of recommended movies that are selected from the test set of the corresponding user. For the remainder of the paper, recall is reported as the average of all the user's individual recalls.

4.4. Splitting the Dataset for Training and Testing / Cold Start and Warm Start

We apply two different test scenarios called the warm-start and the cold-start. The warm-start scenario corresponds to the case where at least one interaction for all the items and the users appear in the training set such that at least some information in the rating matrix is present for all the test users and items. We use the following recipe to create this condition. If the number of interactions for an item is smaller than 5, then all of its interactions are included only in the training set and no interaction for that item exists in the test set. If the number of interactions for an item is larger than 5, a randomly selected $60\% - 20\% - 20\%$ of these interactions are separated for training, validation, and testing respectively.

On the contrary, the cold-start scenario corresponds to the case where some items have all of their associated interactions appeared in the testing set with no associated interactions in the training set. In order to create this case, %20 of the items are randomly chosen as test items and all of their interactions are separated exclusively for the testing set. Similarly, another %20 of randomly chosen items is dedicated as the validation set where all of their interactions are also removed from the training set.

4.5. Simulation Study

First, we conduct a simulation in which a synthetic dataset is generated according to the configuration described in Section 4.2 with 300 users and 500 items. The purposes of this study are: i) to confirm the hypothesis that incorporating side information into the prediction process improves the performance of the recommender system, ii) to assess the performance sensitivity with respect to the sparsity level and iii) to observe the convergence behavior and iv) to test scalability of the proposed model. In this synthetic dataset, five modalities of the information exist including the Gaussian and categorical modalities for the user/item side information as well as the rating matrix. We induce the missing values and calculate the MSE performance on the fully observed rating matrix with respect to the varying fractions of missing values from %0 to %95 with

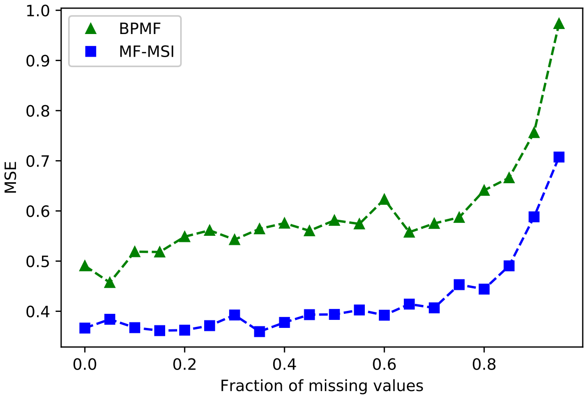
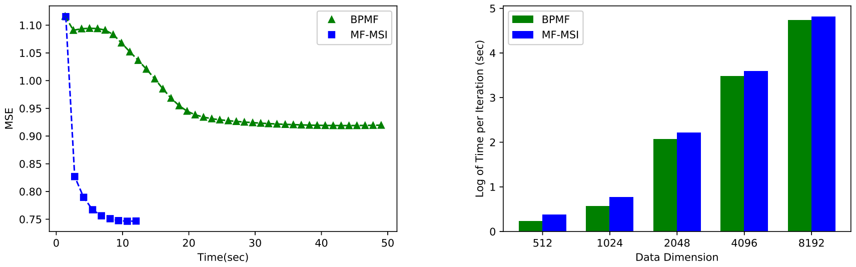


Figure 2: MSE vs missing value fraction on synthetic dataset

increments of %5. In this section, the proposed model is only tested against the BPMF model which does not utilize any side information but is otherwise the same predictor. The results are presented in Figure 2.

530 We see a significant improvement in the performance of the MF-MSI model over the BPMF across all different fractions of missing values which serves as a strong empirical validation of the model and its behavior. These results are averaged across 10 experiments for each fraction. Although the difference is around 0.13 when the missing value fraction is 0, i.e., all ratings are observed, 535 the difference more than doubles to 0.27 when the missing value fraction is %0.95. This indicates that the MF-MSI model fits better to the dataset that follows a generative model assumption even when the fraction of missing values is very high with the help of observed side information. The high fraction of missing values case in this experiment is important and relevant since in most of 540 the real world datasets, the rating matrices have high sparsity ($>$ %95) levels.

Next, we observe the convergence rate of the MF-MSI model compared to the BPMF model for the same hyperparameter configuration. Figure 3.(a) shows the "MSE with respect to time in seconds" when the fraction of missing values is set as %95. The MF-MSI model tends to converge faster to a lower MSE. 545 The BPMF, on the other hand, displays an elbow during the first few iterations



(a) MSE convergence

(b) Log of time per iteration vs data dimension (number of users and items)

Figure 3: Convergence analysis on synthetic dataset

which slows down the convergence. The addition of side information seems to remove this elbow and lead to a smoother and faster (as much as 4x times) convergence.

As discussed in Section 3.3, when the number of users I and the number of items J are large compared to the dimensions of the side information, time complexity reduces to a factor of $I \times J$. Figure 3.b shows the log of time per iteration when I and J increase from 512 to 8192. Although the total time increases exponentially for both models, the time difference between them vanishes as the numbers of users and items increase. As a result, one can conclude that adding the side information by using the proposed approach does not necessarily increase the time complexity per iteration for large scale applications while leading to an almost order of magnitude faster convergence to a more accurate point in a smaller number of iterations.

4.6. Movie Recommendation Study

In this section, we conduct a comprehensive experimental study to assess the performance of the proposed algorithm by using the MovieLens datasets. Firstly, we show the qualitative results by examining the latent space learned by the algorithm. Next, we compare the prediction and ranking performance of the algorithm with the baseline MF, the standard benchmark and the recent state-of-the-art algorithms, respectively. As a pre-processing, the real-valued side information for both the users and the items is normalized to have zero mean and unit variance. The categorical information is converted to 1-of-K binary representations. The last classes are designated as pivots such that M_u and M_v are equal to the number of the corresponding user and item classes minus 1. Additionally, the model hyper-parameters of each algorithm in the experimental study are optimized by using the validation set for each dataset. A grid search is performed and the parameter set that corresponds to the best performance in the validation set is used in the test set to report the evaluation results.

4.6.1. Examining the Latent Space

In this section, we analyze how well the proposed generative model fits the data by illustrating how the users and the items are grouped together in the latent space with respect to their statistical similarities. Since the posterior distributions of the user and the item latent variables are obtained after inference, the latent space can be discovered and explored by using KL divergence, which is a more convenient distance metric than the Euclidean distance due to the availability of posterior covariances. The closeness of two users can be assessed via KL divergence between the two multivariate Gaussians by using the following closed form expression:

$$D_{KL}(\boldsymbol{u}_i||\boldsymbol{u}_k) = \frac{1}{2}\Big(\mathrm{tr}(\boldsymbol{\Sigma}_{\boldsymbol{u}k}^{-1}\boldsymbol{\Sigma}_{\boldsymbol{u}i}) + (\boldsymbol{m}_{\boldsymbol{u}k} - \boldsymbol{m}_{\boldsymbol{u}i})^T\boldsymbol{\Sigma}_{\boldsymbol{u}k}^{-1}(\boldsymbol{m}_{\boldsymbol{u}k} - \boldsymbol{m}_{\boldsymbol{u}i}) - K + \ln\Big(\frac{|\boldsymbol{\Sigma}_{\boldsymbol{u}k}|}{|\boldsymbol{\Sigma}_{\boldsymbol{u}i}|}\Big)\Big),$$

where “tr” denotes the trace operator. Table 2 shows three sample users neighboring in the latent space. First, User1 is selected randomly among all users and then the two closest users are found using the KL divergence. The metadata of the corresponding users is provided in the table along with the names of the movies the users liked and their corresponding metadata. One can observe the similarity of the users’ demographic information such as gender (all female) and age (all middle age) and the fact that the occupations are not contrary. Similarly on the item side, one can see that some movies such as ’Ice Storm’

Table 2: Three sample users from MovieLens 100K that are close in the learned latent space.

User 1 (Female, 40, Librarian)		
Boogie Nights	1997	Drama
In and Out	1997	Comedy
Postman, The	1997	Drama
Mad City	1997	Action, Drama
Ice Storm	1997	Drama
User 2 (Female, 50, Other)		
Ice Storm	1997	Drama
As Good As It Gets	1997	Comedy, Drama
Wings of the Dove, The	1997	Drama, Romance, Thriller
Good Will Hunting	1997	Drama
Wag the Dog	1997	Comedy, Drama
User 3 (Female, 35, Administrator)		
Cold Comfort Farm	1995	Comedy
Postman, The	1997	Drama
Emma	1996	Drama, Romance
Sense and Sensibility	1995	Drama, Romance
George of the Jungle	1997	Childrens, Comedy

and 'The Postman' appear in each user's list. The movie release dates are also close and the genres (such as comedy and romance) are overlapping. It is obvious that the model fits all the preferences from the rating matrix as well as the side information which includes the metadata of the users and the items in the latent space. Furthermore, we can use the similar approach to find movies that are close in the latent space. A sample of two different movie groups is listed in Table 3. The first movie in each list is chosen randomly and the KL divergence is used to find the 4 closest movies in the latent space for each group. Much like the user case, one can see how the side information including the genre and the movie release dates can help improve the clustering performance of mixed data type observations.

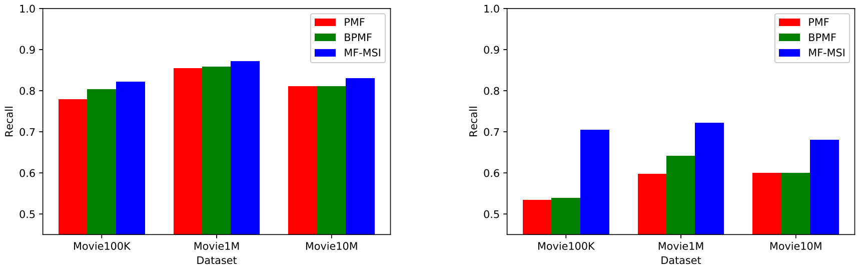
Table 3: Two sample movie groups from MovieLens 100K which are close in the learned latent space.

Group 1		
Shall We Dance?	1937	Comedy, Musical, Romance
Gay Divorcee, The	1934	Comedy, Musical, Romance
Top Hat	1935	Comedy, Musical, Romance
Women, The	1939	Comedy
Band Wagon, The	1953	Comedy, Musical
Group 2		
Ghost	1990	Comedy, Romance, Thriller
Pretty Woman	1990	Comedy, Romance
While You Were Sleeping	1995	Comedy, Romance
In the Line of Fire	1993	Action, Thriller
American President	1995	Comedy, Drama, Romance

4.6.2. Comparison with Baseline MF Algorithms

For all the models in this category, the latent space dimension is chosen the same for a fair comparison, $K = 10$. Prior hyper-parameters λ_u and λ_v are chosen as 1. The rating precision c is initialized as 1. Figure 4.(a) shows the recall performances for both the warm and the cold settings. In the warm setting, as shown in the figure on the left, all models perform similarly (with slightly higher performance for the proposed algorithm) due to the cross-information being included in the rating matrix for both the training and the testing sets. As expected, the PMF model performs the worst in every scenario, while the proposed MF-MSI model performs slightly better than all the other models for all the three datasets.

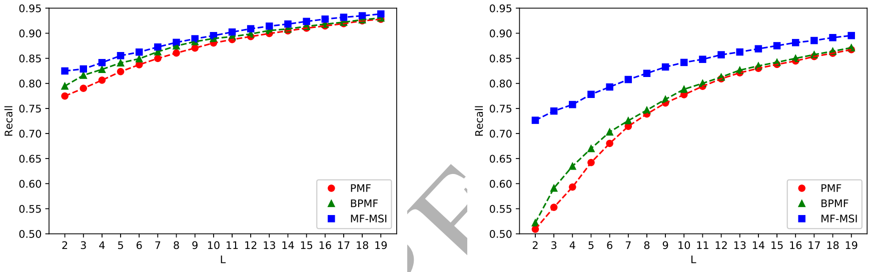
The advantages of the proposed method become much clearer as shown in Figure 4.(b) when the more challenging and realistic cold setting environment is used. As expected, all three algorithms perform worse than the warm setting startup, however, the proposed model is significantly less affected. Naturally, the PMF and the BPMF models cannot generalize properly due to the initial



(a) Warm setting

(b) Cold setting

Figure 4: Recall performances on MovieLens datasets when L = 2



(a) Warm setting

(b) Cold setting

Figure 5: Recall performances for the varying number of recommended movies

lack of information in the rating matrix for the cold setting, which makes their performances depend highly on proper initialization.

Next, we analyze the effect of the number of recommended movies, L, on the recall performance. Figure 5 shows that the differences between the recall performances of the models are getting larger when the number of recommended movies (L) decreases. This behavior is even more apparent in the cold setting case suggesting that the side information allows the proposed model to have a significantly better recommendation performance specifically for its top recommendations. This represents the ideal case for a commercial recommender system as the majority of the users focus on the top 2-3 items in their recommended list where accuracy becomes more important.

4.6.3. Comparison with the Standard Benchmark Algorithms

The standard algorithms are not capable of incorporating any side information. To this end, the comparison is performed only for the warm-start condition via MSE metric. Each experiment is repeated over 10 times to obtain a statistically meaningful result. At each trial, the training/testing split is performed randomly. As indicated in table 4, the proposed model outperforms the standard algorithms by exploiting the side information. The closest performance is achieved by the SVDpp and BPMF. KNN models cannot produce results for the MovieLens 10M since the dataset is too large for these models to process*. Specifically, these models require similarity matrices to be computed and stored in the memory. In the case of the MovieLens 10M dataset, the sizes of the matrices are 71567×71567 and 10681×10681 for the user-based and the item-based models respectively which requires a significantly large memory size for storage and processing. Our 32GB memory could not able to store the matrices. This scalability issue is a well-known bottleneck of neighborhood-based models [47, 3].

4.6.4. Comparison with Recent State-of-the-art Algorithms

The models in this category are able to incorporate the side information. We compare their performances with the proposed model under both the warm and the cold start conditions. For recall performance, the number of recommended movies L is selected as 10. Table 5 presents the averaged results over 10 experiments with different random split initializations for each experiment. It is important to note that, as in the case of the KNN models, the LCE, DecRec and KMF algorithms cannot scale to the Movielens-10M dataset due to the space complexity of the models and their imposed memory limitations*. The LCE and DecRec algorithms require storing the similarity matrices calculated by using the side information of the users and the items, which results in $O(I^2+J^2)$ space complexity. Additionally, the KMF algorithm requires inverting these matrices to obtain kernels for the Gaussian Process priors which results in $O(I^3+J^3)$ computational complexity. LightFM and MF-MSI do not have these types of

Table 4: MSE comparison with standard benchmark algorithms on the warm setting

Algorithms	**MovieLens-100K**	**MovieLens-1M**	**MovieLens-10M**
NormalPredictor	2.304	2.268	1.329
BaselineOnly	0.893	0.828	0.750
KNNBasic	0.964	0.872	*
KNNWithMeans	0.910	0.872	*
KNNWithZScore	0.910	0.874	*
KNNBaseline	0.934	0.901	*
NMF	0.872	0.848	0.766
SVD	0.895	0.794	0.664
SVDpp	0.863	0.767	0.659
SlopeOne	0.893	0.824	0.743
CoClustering	0.949	0.841	0.781
PMF	0.901	0.802	0.691
BPMF	0.850	0.776	0.671
MF-MSI	**0.812**	**0.728**	**0.632**

bottlenecks since they work on the raw features instead of the similarity matrices. LCE also needs a relatively large latent space dimension compared to MF-MSI. The reason is that the side information, which is the similarity matrix, is high dimensional. In order to project this matrix to the latent space, the dimension should be extended significantly. In [28], the authors suggest a latent space dimension of 500 while MF-MSI can achieve similar results with only a 20 dimensional space. That makes the proposed algorithm faster and more efficient compared to LCE. For instance, for the MovieLens 100K dataset, MF-MSI converges in 1.9s which is several orders of magnitude faster than LCE which requires 61.2s for reasonable performance.

In the warm-start scenario, MF-MSI performs better than all the other algorithms in terms of both RMSE and Recall. For the MovieLens 10M dataset, the proposed algorithm outperforms LightFM which is the only other state-of-the-art algorithm in this list scalable to the size of this dataset. In the cold-start

Table 5: Performance comparison with recent state-of-the-art algorithms.

	MovieLens-100K		**MovieLens-1M**		**MovieLens-10M**	
Warm-Start	*MSE*	*Recall*	*MSE*	*Recall*	*MSE*	*Recall*
LCE	0.854	0.890	0.729	0.901	*	*
DecRec	1.187	0.842	1.196	0.814	*	*
KMF	0.863	0.892	0.769	0.900	*	*
LightFM	1.021	0.889	1.034	0.893	1.827	0.886
MF-MSI	**0.812**	**0.900**	**0.728**	**0.904**	**0.632**	**0.902**
Cold-Start						
LCE	1.253	0.849	1.363	0.819	*	*
DecRec	1.234	0.854	1.208	0.823	*	*
KMF	1.208	0.821	**1.140**	0.791	*	*
LightFM	1.274	0.857	1.343	0.822	2.047	0.824
MF-MSI	**1.192**	**0.861**	1.261	**0.827**	**1.490**	**0.886**

scenario, MF-MSI outperforms the other algorithms in terms of ranking performance. However, KMF has better MSE performance for MovieLens 1M. Considering the scalability issue of KMF, one can conclude by the results reported in Table 5 that the overall performance of the proposed model is higher than the competitive algorithms in a significant majority of the test scenarios using the three differently sized datasets.

5. Conclusion

In this paper, we introduce a fundamentally different approach to incorporate multimodal side information using a novel probabilistic generative framework for recommender systems. A scalable and computationally efficient statistical inference method based on variational EM is derived for datasets with very sparse interactions between the users and the items to exploit their associated multimodal side information. The Bayesian structure of the model naturally enables the discovery of full multi-variate distributions over the latent space to provide a better prediction performance in both the mean-square-error and recall metrics as more side information becomes available. The improvement in both the accuracy and the ranking performances of the proposed model is clearly demonstrated over a wide range of popular benchmark models for both warm and cold start test scenarios across both synthetic and real datasets. In fact, the state-of-the-art performance is achieved for the majority of the cases when compared to other recently introduced recommender systems which also incorporate the side information in different ways.

Our findings have several important implications when it comes to the next generation recommender systems. First of all, the fact that side information is utilized to improve the performance suggests that any additional knowledge acquisition by the companies both for the items they are promoting and their users would be beneficial regardless of the platform. Furthermore, the reduction in computational complexity to the level of scaling linearly with both the number of users and the number of items, would allow competitively scalable big data applications even when additional side information is rich and complex in nature.

6. Limitations and Future Research Directions

Ultimately, the main purpose of this paper is to serve as the proof-of-concept for the MF-MSI algorithm supported by both the fundamental derivations and empirical observations with a comprehensive experimental setup. Nonetheless, there are three separate promising pathways for further research. For instance, we demonstrate the scalability of the proposed model in terms of time complexity. While the memory complexity of the MF-MSI algorithm is better, especially when compared to the more recently proposed approaches, it still represents a challenge especially for extremely large datasets. Recent work on variational inference [55] provide promising solutions to such memory problems with stochastic optimization. As future work, stochastic variational EM will be introduced to the model to deal with the memory complexity issues for very large datasets to ultimately support online learning which is necessary for e-commerce applications. As additional future work, the model can incorporate recent techniques in the literature as discussed in Section 2 primarily to increase the prediction performance even further. For instance, local factorization can be performed with a mixture model, or the dimensions of the latent space can be automatically inferred within a mixture rank model. Finally, the current structure of the multimodal side information used in the experimental setup (such as age, occupation, title, year) is comparatively simple compared to the richer and more heterogeneous representations such as textual user reviews, movie synopsis. While the underlying model of inferring the multimodal latent space is capable of expanding to more complex data representations, a significant portion of the future work will nonetheless focus on exploiting more diverse combinations of side information with the help of recent breakthroughs in data fusion and language modeling such as contextual word embeddings.

Declaration of interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

References

References

[1] Y. Shi, M. Larson, A. Hanjalic, Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges, ACM Computing Surveys (CSUR) 47 (1) (2014) 3.

[2] A. Taneja, A. Arora, Recommendation research trends: review, approaches and open issues, International Journal of Web Engineering and Technology 13 (2) (2018) 123–186.

[3] C. C. Aggarwal, et al., Recommender systems, Springer, 2016.

[4] C. M. Bishop, Pattern recognition and machine learning, springer, 2006.

[5] A. Ilin, T. Raiko, Practical approaches to principal component analysis in the presence of missing values, Journal of Machine Learning Research 11 (2010) 1957–2000.

[6] D. M. Blei, A. Kucukelbir, J. D. McAuliffe, Variational Inference: A Review for Statisticians, Journal of the American Statistical Association 112 (518) (2017) 859–877.

[7] A. W. Van der Vaart, Asymptotic statistics, Vol. 3, Cambridge university press, 2000.

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

[9] A. Mnih, R. R. Salakhutdinov, Probabilistic matrix factorization, in: Advances in neural information processing systems, 2008, pp. 1257–1264.

[10] R. Salakhutdinov, A. Mnih, Bayesian Probabilistic Matrix Factorization using MCMC, 25th International Conference on Machine Learning (ICML-2008).

[11] J. Lee, S. Kim, G. Lebanon, Y. Singer, S. Bengio, Llorma: local low-rank matrix approximation, The Journal of Machine Learning Research 17 (1) (2016) 442–465.

[12] K. Wang, W. X. Zhao, H. Peng, X. Wang, Bayesian probabilistic multi-topic matrix factorization for rating prediction, IJCAI International Joint Conference on Artificial Intelligence 2016-Janua (2016) 3910–3916.

[13] W. Ma, Y. Wu, M. Gong, C. Qin, S. Wang, Local Probabilistic Matrix Factorization for Personal Recommendation, Proceedings - 13th International Conference on Computational Intelligence and Security, CIS 2017 2018-Janua (2018) 97–101.

[14] Z. Ghahramani, G. E. Hinton, et al., The em algorithm for mixtures of factor analyzers, University of Toronto.

[15] D. Li, C. Chen, W. Liu, T. Lu, N. Gu, S. Chu, Mixture-rank matrix approximation for collaborative filtering, in: Advances in Neural Information Processing Systems, 2017, pp. 477–485.

[16] S. Zhang, L. Yao, A. Sun, Deep learning based recommender system: A survey and new perspectives, arXiv preprint arXiv:1707.07435.

[17] S. Sedhain, A. K. Menon, S. Sanner, L. Xie, AutoRec : Autoencoders Meet Collaborative Filtering, WWW 2015 Companion: Proceedings of the 24th International Conference on World Wide Web (2015) 111–112.

[18] S. Li, J. Kawale, Y. Fu, Deep collaborative filtering via marginalized denoising auto-encoder, in: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, ACM, 2015, pp. 811–820.

[19] J. Fan, T. Chow, Deep learning based matrix completion, Neurocomputing 266 (2017) 540–549.

[20] D. Agarwal, B.-C. Chen, Regression-based latent factor models, Proceedings of the 15th ACM International Conference on Knowledge Discovery and Data Mining (2009) (2009) 19.

[21] Y.-D. Kim, S. Choi, Scalable variational Bayesian matrix factorization with side information, Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS), 2014.

[22] T. Zhou, H. Shan, A. Banerjee, G. Sapiro, Kernelized probabilistic matrix factorization: Exploiting graphs and side information, in: Proceedings of the 2012 SIAM international Conference on Data mining, SIAM, 2012, pp. 403–414.

[23] Z. Wang, J. Liang, R. Li, Exploiting user-to-user topic inclusion degree for link prediction in social-information networks, Expert Systems with Applications 108 (2018) 143–158.

[24] I. Porteous, A. U. Asuncion, M. Welling, Bayesian matrix factorization with side information and dirichlet process mixtures., in: AAAI, 2010.

[25] C. Wang, D. M. Blei, Collaborative topic modeling for recommending scientific articles, in: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2011, pp. 448–456.

[26] D. Blei, M. Jordan, A. Y. Ng, Latent Dirichlet allocation, Journal of Machine Learning Research 3 (2003) 993–1022.

[27] M. Jamali, M. Ester, A matrix factorization technique with trust propagation for recommendation in social networks, in: Proceedings of the fourth ACM conference on Recommender systems, ACM, 2010, pp. 135–142.

[28] M. Saveski, A. Mantrach, Item cold-start recommendations: learning local collective embeddings, in: Proceedings of the 8th ACM Conference on Recommender systems, ACM, 2014, pp. 89–96.

[29] M. E. Khan, Y. J. Ko, M. Seeger, Scalable collaborative bayesian preference learning, in: Proceedings of the 17th International Conference on Artificial Intelligence and Statistics, Vol. 33, 2014, pp. 475–483.

[30] X. Li, J. She, Collaborative variational autoencoder for recommender systems, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2017, pp. 305–314.

[31] S. Rendle, Factorization machines, in: 2010 IEEE International Conference on Data Mining, IEEE, 2010, pp. 995–1000.

[32] S. Rendle, Z. Gantner, C. Freudenthaler, L. Schmidt-Thieme, Fast context-aware recommendations with factorization machines, in: Proceedings of the

34th international ACM SIGIR conference on Research and development in Information Retrieval, ACM, 2011, pp. 635–644.

[33] A. Taneja, A. Arora, Cross domain recommendation using multidimensional tensor factorization, Expert Systems with Applications 92 (2018) 304–316.

[34] M. M. Khan, R. Ibrahim, M. Younas, I. Ghani, S. R. Jeong, Facebook interactions utilization for addressing recommender systems cold start problem across system domain, Journal of Internet Technology 19 (3) (2018) 861–870.

[35] I. Fernández-Tobías, I. Cantador, P. Tomeo, V. W. Anelli, T. Di Noia, Addressing the user cold start with cross-domain collaborative filtering: exploiting item metadata in matrix factorization, User Modeling and User-Adapted Interaction 1–44.

[36] A. K. Menon, K.-P. Chitrapura, S. Garg, D. Agarwal, N. Kota, Response prediction using collaborative filtering with hierarchies and side-information, in: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2011, pp. 141–149.

[37] M. Kula, Metadata embeddings for user and item cold-start recommendations, CEUR Workshop Proceedings 1448 (2015) 14–21.

[38] A. Elbadrawy, G. Karypis, User-specific feature-based similarity models for top-n recommendation of new items, ACM Transactions on Intelligent Systems and Technology (TIST) 6 (3) (2015) 33.

[39] I. Barjasteh, R. Forsati, D. Ross, A.-H. Esfahanian, H. Radha, Cold-start recommendation with provable guarantees: A decoupled approach, IEEE Transactions on Knowledge and Data Engineering 28 (6) (2016) 1462–1474.

[40] Z. Gantner, L. Drumond, C. Freudenthaler, S. Rendle, L. Schmidt-Thieme, Learning attribute-to-feature mappings for cold-start recommendations., Citeseer.

[41] Z. Zhang, Y. Liu, Z. Zhang, B. Shen, Fused matrix factorization with multi-tag, social and geographical influences for poi recommendation, World Wide Web 1–16.

[42] A. Klami, S. Virtanen, S. Kaski, Bayesian exponential family projections for coupled data sources, arXiv preprint arXiv:1203.3489.

[43] M. E. Khan, G. Bouchard, K. P. Murphy, B. M. Marlin, Variational bounds for mixed-data factor analysis, in: Advances in Neural Information Processing Systems, 2010, pp. 1108–1116.

[44] Y. Yilmaz, A. O. Hero, Multimodal factor analysis, IEEE International Workshop on Machine Learning for Signal Processing, MLSP 2015-Novem (2) (2015) 1–6.

[45] Y. Yılmaz, A. O. Hero, Multimodal event detection in twitter hashtag networks, Journal of Signal Processing Systems 90 (2) (2018) 185–200.

[46] D. Böhning, Multinomial logistic regression algorithm, Annals of the institute of Statistical Mathematics 44 (1) (1992) 197–200.

[47] K. P. Murphy, Machine learning: a probabilistic perspective, 2012.

[48] Y. Koren, Factor in the neighbors: Scalable and accurate collaborative filtering, ACM Transactions on Knowledge Discovery from Data (TKDD) 4 (1) (2010) 1.

[49] F. Ricci, L. Rokach, B. Shapira, Introduction to recommender systems handbook, in: Recommender systems handbook, Springer, 2011, pp. 1–35.

[50] X. Luo, M. Zhou, Y. Xia, Q. Zhu, An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems, IEEE Transactions on Industrial Informatics 10 (2) (2014) 1273–1284.

[51] D. D. Lee, H. S. Seung, Algorithms for non-negative matrix factorization, in: Advances in neural information processing systems, 2001, pp. 556–562.

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

[53] D. Lemire, A. Maclachlan, Slope one predictors for online rating-based collaborative filtering, in: Proceedings of the 2005 SIAM International Conference on Data Mining, SIAM, 2005, pp. 471–475.

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

[55] M. D. Hoffman, D. M. Blei, C. Wang, J. Paisley, Stochastic variational inference, The Journal of Machine Learning Research 14 (1) (2013) 1303–1347.