Frontiers of Computer Science CN 10-1014/TP
https://doi.org/10.1007/s11704-022-1213-7 ISSN 2095-2228
LETTER

A Bayesian Matrix Factorization Model for Dynamic User Embedding in Recommender System

Kaihan ZHANG, Zhiqiang WANG, Jiye LIANG, Xingwang ZHAO

Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China

Front. Comput. Sci., **Just Accepted Manuscript** • 10.1007/s11704-022-1213-7

http://journal.hep.com.cn on February 14, 2022

Front.Comput.Sci.
https://doi.org/10.1007/s11704-022-1213-7

LETTER

A Bayesian Matrix Factorization Model for Dynamic User Embedding in Recommender System

Kaihan ZHANG [1] **Zhiqiang WANG** (✉)[1] **Jiye LIANG** [1] **Xingwang ZHAO** [1]

1 Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China

1 Introduction

The main idea of recommender system is how to learn accurate users' embeddings from behavior data [1]. Each dimension of users' embeddings can reflect the interests of users in different potential aspects. In real-world scenarios, users' interests are drifting over time, which brings a challenge to learn accurate dynamic users' embeddings. Recently, various time-aware recommendation methods have been proposed to tackle this problem by modeling the evolution process of users' interests [2–4]. However, they usually assume that users' embeddings drift with the same range on all dimensions. In practice, users' embeddings should change diversely on different dimensions over time. Specifically, for the rapidly changing interests of the users, the corresponding dimensions should change significantly. On the contrary, the dimensions representing stable interests may change slightly.

To address the above issue, we design a Bayesian Matrix Factorization model for Dynamic user Embedding (BMFDE) in recommender system. Specifically, BMFDE extends the probabilistic matrix factorization model (PMF) to capture the drifting regularities of users' embeddings over time. By introducing the dynamic covariance prior, BMFDE regularizes the users' embeddings at adjacent time intervals to be similar to some extent. In particular, to capture the personalized drifting range for each dimension of users' embeddings, we adaptively learn the precision matrix in dynamic covariance prior from the users' dynamic behavior data. Moreover, a variational expectation maximization algorithm is designed to infer all unknown parameters of the proposed model. The experimental results demonstrate that BMFDE achieves better performance than other comparison methods.

Received month dd, yyyy; accepted month dd, yyyy

E-mail: wangzq@sxu.edu.cn

2 Proposed model

With the consideration of temporal information, at time interval t, we use the inner product of U_i^t and V_j^t to fit the observed rating R_{ij}^t, where $U_i^t \in \mathbb{R}^d$ and $V_j^t \in \mathbb{R}^d$ denote the embeddings of user i and item j, respectively. And d is the number of latent features. The conditional distribution over the observed ratings is defined as:

$$p(R|U,V,\sigma_R^2) = \prod_{t=1}^{s}\prod_{i=1}^{m}\prod_{j=1}^{n} \mathcal{N}(R_{ij}^t|(U_i^t)^{\mathrm{T}}(V_j^t),\sigma_R^2)^{I_{ij}^t}, \quad (1)$$

where I_{ij}^t is the indicator function that is equal to 1 if user i rated item j at time t and equal to 0 otherwise. m and n represent the number of users and items, respectively. Generally, the items are usually described with some fixed attributes [4]. So we assume that items' embeddings are time-invariant, i.e., $V_j^1 = V_j^2 = ... = V_j^s = V_j$.

To model dynamic users' embeddings at different time intervals, we propose a series of dynamic covariance priors to regularize the relevances of users' embeddings between adjacent time intervals. Formally,

$$\begin{aligned} p(U|T,\sigma_U^2) &\propto p(U|\sigma_U^2)\times p(U|T) \\ &= \prod_{t=1}^{s}\prod_{i=1}^{m} \mathcal{N}(U_i^t|0,\sigma_U^2\mathbf{I})\times\mathcal{N}(U_i^t|U_i^{t-1},(T_{U_i}^t)^{-1}), \end{aligned} \quad (2)$$

where $T = \{T_U^1, ..., T_U^s\}$ and $T_U^t = \{T_{U_1}^t, ..., T_{U_m}^t\}$. These priors reflect the drifting process of users' embeddings over time. Specifically, we want to learn $T_{U_i}^t$ to capture the drifting rates for each dimension of U_i from time $t-1$ to t. Since we assume all latent features are independent of each other, $T_{U_i}^t$ is designed as a $d \times d$ diagonal matrix. Besides, a zero-mean Gaussian prior is chosen to avoid over-fitting. Note that at time $t = 1$, we ignore the second term only use zero-mean Gaussian prior. Moreover, we choose a conjugate prior

Wishart distribution $\mathcal{W}$ for $T^t_{U_i}$,

$$p(T|\nu_U, \Lambda_U) = \prod_{t=1}^{s}\prod_{i=1}^{m} \mathcal{W}(T^t_{U_i}|\nu_U, \Lambda_U), \tag{3}$$

where ν_U denotes the degree of freedom and Λ_U is the scale matrix. We fix $\nu_U = d$ and learn Λ_U from data.

Following Eqs.(1)-(3), the posterior distribution needed to be optimized in our model is:

$$\begin{aligned}&\arg\max_{U,V,T} \prod_{t=1}^{s}\prod_{i=1}^{m}\prod_{j=1}^{n} \mathcal{N}(R^t_{ij}|(U^t_i)^{\mathrm{T}}V_j, \sigma_R^2)^{I^t_{ij}}\\ &\prod_{t=1}^{s}\prod_{i=1}^{m}[\mathcal{N}(U^t_i|U^{t-1}_i, (T^t_{U_i})^{-1})\mathcal{W}(T^t_{U_i}|\nu_U, \Lambda_U)\\ &\mathcal{N}(U^t_i|0, \sigma_U^2\mathbf{I})]\prod_{j=1}^{n}\mathcal{N}(V_j|0, \sigma_V^2\mathbf{I}).\end{aligned} \tag{4}$$

Approximate Inference. In order to infer the unknown parameters in Eq. (4), we adopt mean-field variational expectation maximization algorithm to optimize the hidden variables $Z = \{U, V, T\}$ and model parameters $\theta = \{\sigma_R^2, \sigma_U^2, \sigma_V^2, \Lambda_U\}$ iteratively. Concretely, we apply a tractable auxiliary probability $q(Z|\theta')$ to approximate the posterior distribution $p(Z|R, \theta)$, where $\theta' = \{\lambda^t_{U_i}, \gamma^t_{U_i}, \nu^t_{U_i}, \Lambda^t_{U_i}, \lambda_{V_j}, \gamma_{V_j}\}$. Then, we adopt variational expectation maximization to iteratively update model parameters θ and variational parameters θ'.

In variational E-step, we focus on the optimization of parameters θ', where θ are fixed. We consult the formula of the logarithmic form of multivariate normal distribution and then obtain the updating rules of $\gamma^t_{U_i}$ and $\lambda^t_{U_i}$:

$$\begin{aligned}\gamma^t_{U_i} =&[\frac{1}{\sigma_R^2}\sum_{j=1}^{n} I^t_{ij}(\lambda_{V_j}\lambda_{V_j}^{\mathrm{T}} + \gamma_{V_j}) + (d+1)\Lambda^t_{U_i}\\ &+ (d+1)\Lambda^{t+1}_{U_i} + \sigma_U^{-2}\mathbf{I}]^{-1},\\ \lambda^t_{U_i} =&\gamma^t_{U_i}[\frac{1}{\sigma_R^2}\sum_{j=1}^{n} I^t_{ij}R^t_{ij}\lambda_{V_j} + (d+1)\Lambda^t_{U_i}\lambda^{t-1}_{U_i}\\ &+ (d+1)\Lambda^{t+1}_{U_i}\lambda^t_{U_i}].\end{aligned} \tag{5}$$

In the similar way, the updating formulas of γ_{V_j}, λ_{V_j}, $\nu^t_{U_i}$ and $\Lambda^t_{U_i}$ can be obtained:

$$\begin{aligned}\gamma_{V_j} &= [\frac{1}{\sigma_R^2}\sum_{t=1}^{s}\sum_{i=1}^{m} I^t_{ij}(\lambda^t_{U_i}(\lambda^t_{U_i})^{\mathrm{T}} + \gamma^t_{U_i}) + \sigma_V^{-2}\mathbf{I}]^{-1},\\ \lambda_{V_j} &= \gamma_{V_j}[\frac{1}{\sigma_R^2}\sum_{t=1}^{s}\sum_{i=1}^{m} I^t_{ij}R^t_{ij}\lambda^t_{U_i}],\\ \nu^t_{U_i} &= \nu_U + 1,\\ \Lambda^t_{U_i} &= [(\lambda^t_{U_i} - \lambda^{t-1}_{U_i})(\lambda^t_{U_i} - \lambda^{t-1}_{U_i})^{\mathrm{T}} + \gamma^t_{U_i} + \gamma^{t-1}_{U_i} + \Lambda_U^{-1}]^{-1}.\end{aligned} \tag{6}$$

In variational M-step, variational parameters θ' are fixed and model parameters θ need to be updated. Concretely, the optimal values of parameters θ can be written as a closed-form solution. For example, we use $\mathcal{L}$ to denote the lower bound of the log $p(R|\theta)$, and the closed-form solution of σ_R^2 can be computed by letting $\frac{\partial\mathcal{L}}{\partial\sigma_R^2} = 0$:

$$\begin{aligned}\sigma_R^2 =&\frac{1}{|R|}\sum_{t=1}^{s}\sum_{i=1}^{m}\sum_{j=1}^{n} I^t_{ij}[(R^t_{ij})^2 - 2R^t_{ij}(\lambda^t_{U_i})^{\mathrm{T}}\lambda_{V_j} + tr(\gamma^t_{U_i}\gamma_{V_j})\\ &+ ((\lambda^t_{U_i})^{\mathrm{T}}\lambda_{V_j})^2 + (\lambda^t_{U_i})^{\mathrm{T}}\gamma_{V_j}\lambda^t_{U_i} + \lambda_{V_j}^{\mathrm{T}}\gamma^t_{U_i}\lambda_{V_j}].\end{aligned} \tag{7}$$

Similarly, through the derivation we can obtain the solutions of σ_U^2, σ_V^2 and Λ_U:

$$\begin{aligned}\sigma_U^2 &= \frac{1}{smd}\sum_{t=1}^{s}\sum_{i=1}^{m}(\|\lambda^t_{U_i}\|_2^2 + tr(\gamma^t_{U_i})),\\ \sigma_V^2 &= \frac{1}{nd}\sum_{j=1}^{n}(\|\lambda_{V_j}\|_2^2 + tr(\gamma_{V_j})),\\ \Lambda_U &= \frac{1}{(s-1)md}\sum_{t=2}^{s}\sum_{i=1}^{m}(d+1)\Lambda^t_{U_i}.\end{aligned} \tag{8}$$

Rating Prediction. After obtaining the learned parameters by Eqs. (5)-(8), the prediction of the unobserved rating R^t_{ij} is computed by letting $\frac{\partial\mathcal{L}}{\partial R^t_{ij}} = 0$. The solution can be obtained in a closed form as follows:

$$\hat{R}^t_{ij} = (\lambda^t_{U_i})^{\mathrm{T}}\lambda_{V_j}.$$

3 Experiments

Three real-world datasets from MovieLens and Dianping are used in our experiments: 100k, 10m, and dianping. The statistics of these datasets are described in Table 1. Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) are used to measure the performance of all methods. These metrics are widely used in the recommender system.

Table 1 Statistics of datasets

Features	100k	10m	dianping
#users	943	69,878	37,081
#items	1,682	10,677	28,520
#ratings	100,000	10,000,054	1,077,845
density	6.3047%	1.3403%	0.1019%

To evaluate the proposed model, we compare it with matrix factorization-based models PMF, timeSVD++ [2], TMF [3], PCCF [4], and neural network-based model LightGCN [5]. Tables 2-3 report the MAE and RMSE results on three datasets. Fig. 1 shows the training time of these models. We can find that BMFDE achieves the best results than matrix factorization-based methods. Compared with neural network-based model LightGCN, though the performance of BMFDE is a little worse in most cases, the training time of BMFDE is significantly superior to LightGCN.

We investigate the influence of the number of time intervals s and the number of latent features d. We tune s from 1 to 10 by step 1 and search d in $\{1, 3, 5, 10, 15, 20, 25, 30\}$. Fig.

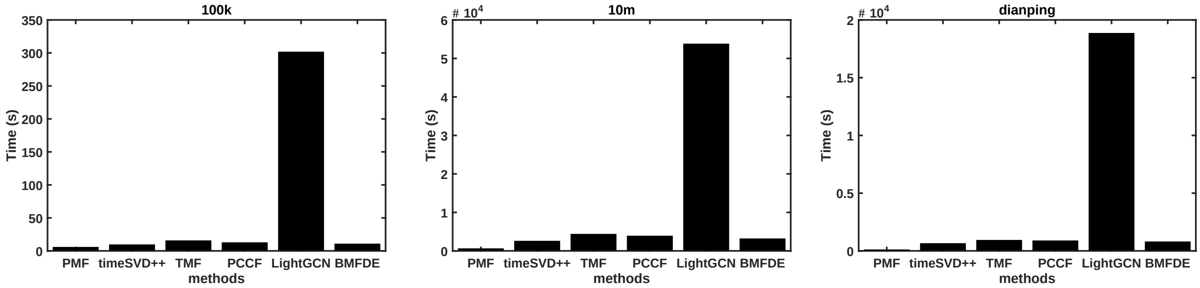

Fig. 1 Training time (seconds) of the comparison methods on three datasets

Table 2 The results compared with matrix factorization-based models

	100k		10m		dianping	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
PMF	0.7414	0.9500	0.6125	0.7991	0.6531	0.8777
timeSVD++	0.7318	0.9516	0.6163	0.8147	0.5999	0.7901
TMF	0.7676	0.9852	0.6118	0.8000	0.5599	0.7608
PCCF	0.7361	0.9351	0.6116	0.7970	0.5892	0.7740
BMFDE	**0.7182**	**0.9177**	**0.6055**	**0.7904**	**0.5427**	**0.7225**

Table 3 The results compared with neural network-based model

	100k		10m		dianping	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
LightGCN	0.7234	0.9270	**0.5993**	**0.7890**	**0.5312**	**0.7039**
BMFDE	**0.7182**	**0.9177**	0.6055	0.7904	0.5427	0.7225

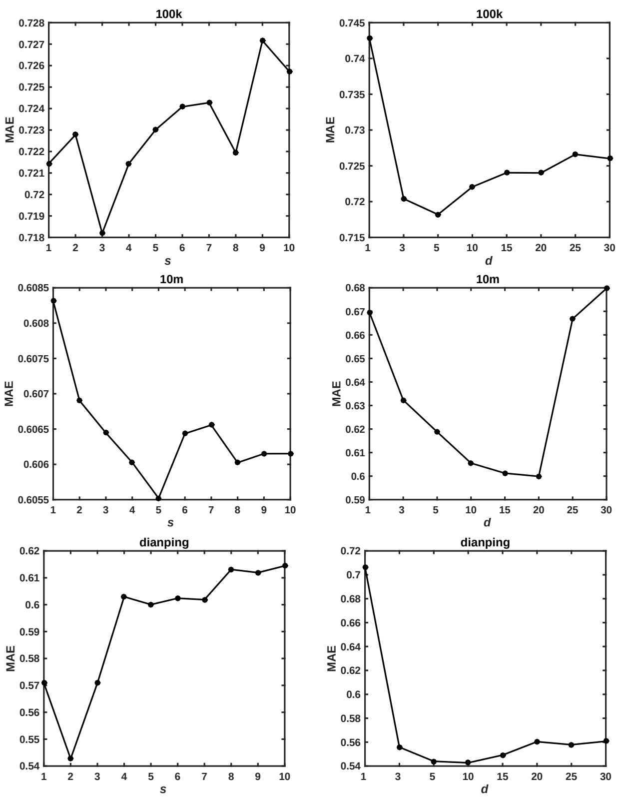

Fig. 2 The influence of parameters s and d on MAE on three datasets

2 reports the results on MAE. We can observe that too larger s and d may cause the overfitting problem and the model performance is adversely affected. The proper s and d have enough ability to capture the drifting regularities of users' interests. Therefore, we set s for 100k, 10m, and dianping to 3, 5, and 2, respectively. And we set d for 100k, 10m, and dianping to 5, 20, and 10, respectively.

4 Conclusion

A Bayesian matrix factorization model for dynamic user embedding has been proposed in this paper. By introducing the dynamic covariance prior, BMFDE can learn the personalized drifting regularities of users' interests from behavior data. Besides, we design an efficient Bayesian inference algorithm to infer unknown parameters of BMFDE. Extensive experiments on real-world datasets demonstrate the effectiveness of our proposed method.

Acknowledgements This work was supported by the National Natural Science Foundation of China (No.61906111, U21A20473, 72171137), the Projects of Key Research and Development Plan of Shanxi Province (201903D121162), and the 1331 Engineering Project of Shanxi Province.

References

1. Zhao Z Y, Zhou H, Li C, Tang J, Zeng Q T. Deepemlan: deep embedding learning for attributed networks. Information Sciences, 2021, 543:382-397
2. Koren Y. Collaborative filtering with temporal dynamics. In: Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2009, 447–456
3. Zhang C Y, Wang K, Yu H K, Sun J L, Lim E P. Latent factor transition for dynamic collaborative filtering. In: Proceedings of SIAM International Conference on Data Mining. 2014, 452–460
4. Guo G B, Zhu F D, Qu S L, Wang X W. Pccf: periodic and continual temporal co-factorization for recommender systems. Information Sciences, 2018, 436:56–73
5. He X N, Deng K, Wang X, Li Y, Zhang Y D, Wang M. Lightgcn: simplifying and powering graph convolution network for recommendation. In: Proceedings of International ACM SIGIR Conference on Research and Development in Information Retrieval. 2020, 639-648