Variational Autoencoder with Stochastic Masks to Solve Exposure Bias in Recommendation

Yingshuai Kou*†‡, Neng Gao*‡, Yifei Zhang*‡
*State Key of Laboratory of Information Security, Chinese Academy of Sciences, Beijing, China
†School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
‡Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
{kouyingshuai, gaoneng, }@iie.ac.cn

***Abstract*—In the recommended scenario, an unobserved interaction may exists in two cases: the user is not interested in the item or the user is not aware of the item at all. This phenomenon leads to a serious exposure bias problem in recommendation system. To solve this problem, we propose Masked Variational Autoencoder (MVAE). Firstly, we predict the missing values in the sparse user-item interaction matrix by matrix completion. Then we randomly mask the elements in the obtained matrix and use them as the input to the variational autoencoder. The decoder can reconstruct the user interaction matrix closer to the true distribution and fully exploit the potential preferences of users in uninteracted items. In particular, we use a combined dual VAE to tackle the exposure bias problem from the user side and the item side respectively. Extensive experiments on three real-world datasets also illustrate the effectiveness of MVAE for solving exposure bias in recommendation.**

***Index Terms*—Recommendation System, Exposure Bias, Variational Autoencoder**

I. Introduction

Recommendation system is a subfield of software engineering that uses machine learning, data mining and other technologies to actively show users items they might like during their browsing interactions[1]. However, user interaction data is observation-based, the full picture of the data is unknown. This leads to a serious exposure bias problem. In the actual scenario, the unobserved sample may exist in two cases, i.e., the item does not match the user's interest or the user doesn't know the item at all. After clarifying the causes of exposure bias, we need to figure out why the existing methods do not work well to solve the exposure bias problem.

a) Deficiencies in model evaluation: Most Recommendation methods [5] is optimized by randomly sampling among uninteracted items as negative examples and maximizing the distance between positive and negative examples. However, as mentioned above, this optimization approach undoubtedly exacerbates the exposure bias.

b) Models lack the ability to generate: Traditional self-supervised approach such as Autoencoder model[12] reconstructs the data by encoding and decoding operations. But the users and items are directly mapped into fixed hidden vectors, the models can't explore the hidden potential preferences of uninteracted users sufficiently.

c) Unable to handle missing values in sparse data: Exposure bias is mainly due to sparse data. Even the generative model is still insufficient to mine users' potential interest preferences from a large number of uninteracted samples.

Hence, we propose the Masked Variational Autoencoder (MVAE) method to solve the exposure bias problem in the recommendation. The origin interaction matrix is first complemented by the Singular Value Decomposition(SVD). SVD maps the users and items into a same vector space and complements the missing values in the matrix by constructing links between user items. We mask a random part of pseudo-matrix and reconstruct the missing elements by VAE. With the generalization capability of masking and the generation capability of VAE, we expect to be able to restore the interaction matrix which is closer to the true distribution. In particular, we design a combined dual VAE structure that combines the exposure of items to the user on the user side and the display of items from the item side.

In summary, our contributions in this paper are as follows:

- To solve the problem that the uninteracted samples contain a large number of potential user preferences due to data sparsity, we use the SVD method to complete the matrix, and the obtained matrix is randomly masked to be the input of VAE.
- We propose a combined dual VAE structure, the user side is used to predict users' potential items of interest, while the item side is used to expose items to interested users, the two-way design effectively increases the robustness of the model.
- We conduct experiments on three real-world recommendation scenarios, and the experimental results show that our method has a better effect of exposure debiasing.

II. Related Work

Research methods for exposure bias can be divided into two categories: debiasing in evaluation and debiasing in training.

The main approach to eliminate exposure bias in the evaluation phase is to use the propensity score. For example, SNIPS [11] evaluates the error of implicit feedback data on traditional metrics such as AUC, DCG@k, Recall@k, and later uses an inverse propensity score framework to offset the exposure bias. The current mainstream approaches try to solve this problem during the training phase of the model. One of them

The corresponding author is Neng Gao.

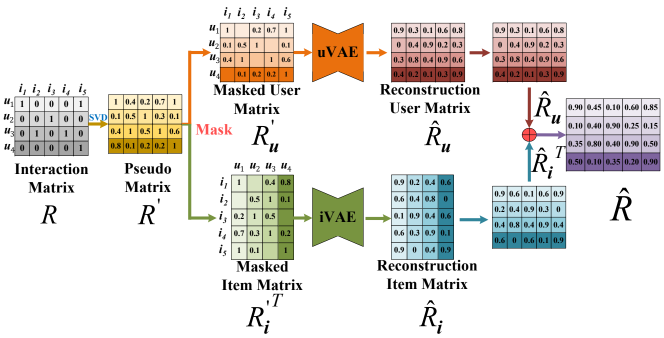

Fig. 1. The framework of MVAE, the original interaction matrix is first complemented by SVD to obtain a pseudo-matrix; the complemented matrix is then masked and used as input to user-based VAE and item-based VAE respectively, and the weighted sum of the reconstructed matrices output by the two decoders is used as the final prediction scoring matrix.

is based on heuristic strategies. For example, user-item feature similarity [6] has also been used to define the confidence level. Another type of research route is resampling, some researchers have explored the use of auxiliary information to augment the sampler. SamWalker++ [9] adds social networks as auxiliary information to the sampling distribution of the model. Although such methods achieve some debiasing effect, the addition of auxiliary information increases the data processing load and the computational complexity of the model.

III. Problem Formulation

We start by characterizing the recommendation problem and introducing notation to be used throughout. In a classic implicit recommendation scenario, we denote the set of users as $\mathcal{U} = \{u_1, u_2, ..., u_n\}$, and the set of items as $\mathcal{I} = \{i_1, i_2, ..., i_m\}$. The history feedback from users to items can be represented as $\mathcal{O} = \{u, i, R_{ui}\}$, where R_{ui} indexes the rating from the user u to the item i. Based on this interaction set $\mathcal{O}$, we can construct an user-item interaction matrix $R \in \{0,1\}^{n\times m}$, where $R_{ui} = 1$ if the interaction of a user u and an item i is observed. The rest of the unobserved samples are denoted by 0 as $R_{ui} = 0$. The intention of the recommendation system is to predict whether a user is interested in a candidate item that he has not interacted with before. Our goal is to learn a prediction function to calculate the probability that user u will click item i.

IV. Methodology

In this section, we illustrate the proposed MVAE in detail. The overall architecture of the model is shown in Fig. 1.

A. Matrix Completion

First of all, we use matrix factorization to complete the matrix. That is, we fill the elements of the matrix that have not generated records (unobserved data) based on the existing data in the matrix (observed data). Here we choose Singular Value Decomposition (SVD) as the matrix completion method. Interaction matrix R can be linearly combined by two smaller matrices according to the SVD principle.

$$R = PQ^T \tag{1}$$

here the original interaction matrix is decomposed into users features matrix P and item features Matrix Q. In the decomposed matrix, each user and item is represented as a feature vector consisting of K features f. The P and Q matrices are obtained by training in the learning process, and a loss function is defined as:

$$E = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{m} I_{ij}(R_{ij} - p(P_i, Q_j))^2 + \frac{k_i}{2}\sum_{i=1}^{n}\|P_i\|^2 + \frac{k_j}{2}\sum_{j=1}^{m}\|Q_j\|^2 \tag{2}$$

where $p(P_i, Q_j)$ represents the prediction score of user i and item j. The prediction function p is the dot product $p(P_i, Q_j) = P_i Q_j^T$. $I \in \{0,1\}^{n\times m}$ in Eq. (3) is an indicator of whether the corresponding position has a interaction. The last two terms on the right side of the equation are the regularization terms to prevent overfitting. After training, we can complete the missing values in the original matrix by the product of matrices P and Q. However, the value of the complement here is based on predictions, which is still not the exact true value. We temporarily call it the pseudo-matrix R'.

B. Stochastic Masks

We mask the pseudo-matrix R' in order to further increase the generalization capability of the model. To be specific, we divide the matrix into submatrices at random in a certain ratio. Then we sample one of the submatrices and mask the remaining submatrices. Here we directly set the values of these positions to 0. Our sampling strategy is simple: we randomly sample the elements of the matrix without replacement according to a uniform distribution. Random sampling on the one hand largely eliminates redundancy, and on the other hand guides the model in learning which values are true in our pseudo-matrix R'.

C. Variational Autoencoder

The variational autoencoder(VAE) reconstructs the matrix by mapping the original data into a standard normal distribution and desampling a representation from the distribution for decoding. For ease of exposition, the input is uniformly denoted by x. In terms of a probabilistic model, the encoder can also be defined by the parametrized posterior distribution $p(z|x)$. Here we use the SVD to complement the original matrix and mask some of the values to obtain a new $\overline{x}$. Then the encoder fits $p(z|\overline{x})$ based on $\overline{x}$. The value of the matrix complement guides us to discover which of the unobserved samples are potentially positive, and by assigning values to them we obtain an $\overline{x}$ distribution that is closer to the true distribution. However, since the posterior probability distribution is difficult to solve in probabilistic models, for each data point we need to approximate the intractable posterior distribution $p(z|\overline{x})$. Therefore, we need to use variational inference [3]. Variational inference maximizes the approximation of the original posterior distribution by a simple variational distribution $q(z)$. We set $q(z)$ to be a fully factorized (diagonal) Gaussian distribution:

$$q(z) = \mathcal{N}\left(\mu, \operatorname{diag}\left\{\sigma^2\right\}\right) \tag{3}$$

Then the distribution is parametrized by θ with both multivariate functions $\mu_\theta(\overline{x})$ and $\sigma^2_\theta(\overline{x})$ being K-vectors and sets the variational distribution as follows:

$$q_{\boldsymbol{\theta}}(z,\overline{x})=\mathcal{N}\left(\mu_{\boldsymbol{\theta}}(\overline{x}),\sigma^2_{\boldsymbol{\theta}}(\overline{x})\right) \tag{4}$$

The encoder of VAE uses the masked $\overline{x}$ as input and derives the corresponding variational parameters of the variational distribution $q_{\boldsymbol{\theta}}(z,\overline{x})$ by the inference model.

VAE's decoder model can be viewed as a generative model $p_\xi(x|z)$, sampling a z from the distribution fitted by the encoder and then reconstructing the original vector x. According to the method of learning latent variable models by variational inference, we can calculate the Evidence Lower Bound (EBLO) of the data, that is, the final derivation in Eq. (5). We use it as the objective to seek maximization of the VAE reconstruction results.

$$\log p\left(\mathrm{x};\xi\right)\geq E_{q_\theta(\mathrm{z}|\overline{\mathrm{x}})}\left[\log p_\xi\left(\mathrm{x}|\mathrm{z}\right)\right]-\mathrm{KL}\left(q_\theta\left(\mathrm{z}\right)\|p\left(\mathrm{z}\right)\right) \tag{5}$$

Then we use the $reparametrization\ trick$ by introducing the regularization hyperparameter β to control the trade-off between the regularization term (i.e., KL loss) and the reconstruction loss. Thus the loss function of the VAE is as follows:

$$\mathcal{L}_\beta\left(\mathrm{x};\xi,\theta\right)=E_{q_\theta(\mathrm{z}|\overline{\mathrm{x}})}[\log p_\xi\left(\mathrm{x}\mid\mathrm{z}\right)]-\beta\cdot\mathrm{KL}\left(q_\theta\left(\mathrm{z}\mid\overline{\mathrm{x}}\right)\|p\left(\mathrm{z}\right)\right) \tag{6}$$

VAE makes flexible use of variational inference, and generates data to answer relevant new questions. The ability to generate inference in this way can help us to better address the problem of exposure bias by exploring the missing values in the interaction matrix together with the mask operation.

D. Combined uVAE and iVAE

We construct a combined model of user-based VAE(uVAE) and item-based VAE(iVAE). The joint optimization of these two VAEs contributes to their fine-tuned calibration, and together uVAE and iVAE can learn complementary information from the user's interaction with the item. For the user-item interaction matrix R, the uVAE reconstructs the matrix row-by-row, while the item VAE reconstructs it column-by-column. The final predicted output:

$$\hat{R}=\alpha\hat{R}_u+(1-\alpha)\hat{R}_i \tag{7}$$

where $\hat{R}_u$ and $\hat{R}_i$ are the uVAE's and iVAE's output reconstruction matrices, respectively. Here we follow the experimental setup of the joint method described above, taking $\alpha=0.5$. According to Eq. (9), we add the loss functions of uVAE and iVAE as the total loss function of our MVAE model:

$$L_{\mathrm{MVAE}}(\mathrm{R}\mid\boldsymbol{\xi},\beta)=\sum_{u\in U}L_{\mathrm{VAE}}\left(\mathrm{R}_u\mid\boldsymbol{\xi}_u,\beta\right)+\sum_{i\in I}L_{\mathrm{VAE}}\left(\mathrm{R}_i^T\,|\,\boldsymbol{\xi}_i,\beta\right) \tag{8}$$

where ξ_u and ξ_i represent the model parameters of uVAE and iVAE, respectively. It should be noted that, unlike the traditional implicit feedback recommendation algorithms, we do not use the pairwise BPR ranking loss which is optimized by maximizing the distance between positive and negative examples. As mentioned above, the negative cases randomly sampled from the uninteracted samples are not necessarily negative, so the loss will inadvertently increase the exposure bias. We use the self-supervised VAE model and optimize the reconstruction loss in a way that can effectively avoid this phenomenon.

V. Experiments

In this section, we evaluate our proposed model MVAE and present its performance on three real-world datasets.

TABLE I
STATISTICS OF THE DATASETS

	Movielens1M	Yelp	Pinterest
#Users	6,027	12,705	55,187
#Items	3,062	9,245	9,911
#Interaction	574,026	318,314	1,500,806
#Sparsity	96.89%	99.73%	99.73%

A. Datasets

To evaluate the effectiveness of our model, We conduct experiments based on three real-world datasets. Basic statistics of the datasets are summarized in Table I. Movielens1M is a widely used benchmark dataset in movie recommendations, Yelp is a subset of Yelp's businesses, Pinterest is an image based content social networking site. Following the previous work [2], [13], we keep only users and items with at least 20 interactions to single out good quality data from the original dataset by using above setting.

B. Baselines

To testify the effectiveness of our method, we compare MVAE against state-of-the-art methods: **BPR**[8] is a matrix decomposition-based model with a pair-wise ranking loss. **CDAE**[10] is a neural network version of SVD. **NCF** [5] combines a neural version of the MF model (GMF) with a multilayer perceptron (MLP) model. **Mult-VAE**[7] extends variational autoencoder (VAE) to collaborative filtering with implicit feedback. **FAWMF**[4] proposes a fast adaptive weighting matrix decomposition based on a variational autoencoder in order to achieve adaptive weight assignment for exposure debiasing. **JCA** [13] proposes a joint collaborative autoencoder framework that learns both user-user and item-item correlations. **JoVA** [2] is a variant of JCA and uses the generative power of VAE for effective exposure debiasing.

C. Evaluation Protocols and Parameters

We utilize two commonly-used metrics to assess the quality of predicted ranked list for each user u: $F1\text{-}Score@K$ and $NDCG@K$. We report the average of these metrics (over testing users). We optimize our model with Adam, and set the learning rate to 0.003. For the training data of each batch, we decompose the matrix into 1500×1500 submatrices. The hyperparameters were set as: $\beta=0.15,\alpha=0.5$. The mask ratio is 4% for each epoch. For each encoder and decoder in the VAE, we have two hidden layers, each of dimension size

TABLE II
OVERALL PERFORMANCE ON MOVIELENS1M, YELP AND PINTEREST

Method	Movielens1M						Yelp						Pinterest					
	F1-score			NDCG			F1-score			NDCG			F1-score			NDCG		
	@1	@5	@10	@1	@5	@10	@1	@5	@10	@1	@5	@10	@1	@5	@10	@1	@5	@10
BPR	.041	.129	.170	.284	.255	.243	.007	.018	.022	.017	.022	.030	.012	.029	.033	.033	.031	.041
CDAE	.052	.147	.187	.343	.290	.273	.016	.032	.036	.038	.039	.047	.015	.035	.040	.042	.039	.051
NCF	.051	.149	.188	.296	.273	.271	.015	.033	.035	.037	.039	.050	.012	.031	.038	.038	.035	.048
Mult-VAE	.052	.142	.180	.343	.289	.270	.015	.032	.034	.035	.038	.047	.015	.035	.040	.047	.040	.050
FAWMF	.060	.166	.207	.378	.318	.299	.015	.029	.031	.036	.036	.043	.013	.031	.036	.042	.036	.045
JCA	.060	.163	.208	.370	.313	.298	.016	.035	.038	.041	.044	.054	.015	.038	.046	.045	.042	.056
JoVA	.061	.167	.212	.372	.314	.301	**.020**	**.039**	.040	**.045**	.048	.058	.020	.047	.054	.058	.052	.064
MVAE	**.063**	**.170**	**.215**	**.383**	**.320**	**.307**	.019	**.039**	**.042**	.043	**.050**	**.060**	**.024**	**.051**	**.059**	**.063**	**.055**	**.070**

320, and the activation function used is $tanh$. the dimension of the potential space d is set to 80. the final output layer uses the sigmoid activation function. We train all the models on a single NVIDIA Tesla A100 GPU for 200 epochs.

D. Overall Performance

Table II reports $F1\text{-}Score$ and $NDCG$ for all datasets and methods. Each metric is averaged across all test users. MVAE significantly outperform the baselines across datasets and metrics. On the Movielens1M dataset, we have made some significant improvements over the optimal baseline algorithm JOVA, especially in the $NDCG@K$ evaluation metric. This is mainly because MVAE takes the masked complementary matrix as input and assign different propensity scores to those potential points of interest. The improvement on the Pinterest dataset was significant, with the best metric improving by 9.375% ($NDCG@10$). The minimum improvement is 5.769%($NDCG@5$). However, we also note that the result of $K = 1$ metric is sub-optimal on the Yelp dataset. Probably because the number of items in Yelp is too large and it is not as efficient when exposing unseen items. But as K increases, our effect improves and becomes optimal. This also suggests that our model gradually increases the exposure of items, which will further improve the recommendation results as the feedback loop of the recommendation system progress. Overall, these results indicate that MVAE achieves better exposure debiasing results than traditional methods and significantly improves the accuracy of the recommendations.

E. Study on MVAE Structure

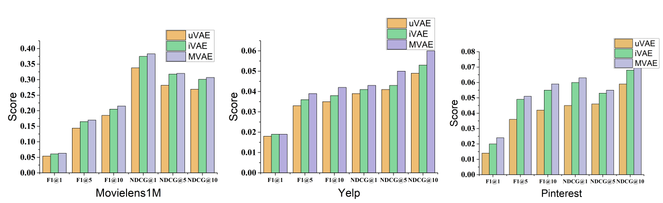

Fig. 2. Test performance of uVAE, iVAE and MVAE in three datasets.

In this section, we conduct ablation studies on uVAE and iVAE separately to demonstrate that our combined model is effective. Fig. 2 illustrates the results. From the experimental results, it can be seen that MAVE improves significantly over both uVAE and iVAE by coupling the dual VAEs from the user side and the item side. It indicates that uVAE and iVAE generate different recommendations which are complementary to each other.

VI. CONCLUSION

In this work, we propose a dual Variational Autoencoder with stochastic masks to solve exposure bias in recommendation. Experiments on three real world datasets confirm the effectiveness of our model.

REFERENCES

[1] Alhijawi, B., Kilani, Y.: The recommender system: a survey. Int. J. Adv. Intell. Paradigms **15**(3), 229–251 (2020)

[2] Askari, B., Szlichta, J., Salehi-Abari, A.: Variational autoencoders for top-k recommendation with implicit feedback. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 2061–2065 (2021)

[3] Blei, D.M., Kucukelbir, A., McAuliffe, J.D.: Variational inference: A review for statisticians. Journal of the American statistical Association **112**(518), 859–877 (2017)

[4] Chen, J., Wang, C., Zhou, S., Shi, Q., Chen, J., Feng, Y., Chen, C.: Fast adaptively weighted matrix factorization for recommendation with implicit feedback. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 3470–3477 (2020)

[5] He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S.: Neural collaborative filtering. In: Proceedings of the 26th international conference on world wide web. pp. 173–182 (2017)

[6] Li, Y., Hu, J., Zhai, C., Chen, Y.: Improving one-class collaborative filtering by incorporating rich user information. In: Proceedings of the 19th ACM international conference on Information and knowledge management. pp. 959–968 (2010)

[7] Liang, D., Krishnan, R.G., Hoffman, M.D., Jebara, T.: Variational autoencoders for collaborative filtering. In: Proceedings of the 2018 world wide web conference. pp. 689–698 (2018)

[8] Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: Bpr: Bayesian personalized ranking from implicit feedback. In: Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence. pp. 452–461 (2009)

[9] Wang, C., Chen, J., Zhou, S., Shi, Q., Feng, Y., Chen, C.: Samwalker++: recommendation with informative sampling strategy. IEEE Transactions on Knowledge and Data Engineering (2021)

[10] Wu, Y., DuBois, C., Zheng, A.X., Ester, M.: Collaborative denoising auto-encoders for top-n recommender systems. In: Proceedings of the ninth ACM international conference on web search and data mining. pp. 153–162 (2016)

[11] Yang, L., Cui, Y., Xuan, Y., Wang, C., Belongie, S., Estrin, D.: Unbiased offline recommender evaluation for missing-not-at-random implicit feedback. In: Proceedings of the 12th ACM conference on recommender systems. pp. 279–287 (2018)

[12] Zhao, X., Ren, Y., Du, Y., Zhang, S., Wang, N.: Improving item cold-start recommendation via model-agnostic conditional variational autoencoder. arXiv preprint arXiv:2205.13795 (2022)

[13] Zhu, Z., Wang, J., Caverlee, J.: Improving top-k recommendation via jointcollaborative autoencoders. In: The World Wide Web Conference. pp. 3483–3482 (2019)