One-Class Collaborative Filtering with the Queryable Variational Autoencoder

Ga Wu*
University of Toronto
wuga@mie.utoronto.ca

Mohamed Reda Bouadjenek
University of Toronto
mrb@mie.utoronto.ca

Scott Sanner*
University of Toronto
ssanner@mie.utoronto.ca

ABSTRACT

Variational Autoencoder (VAE) based methods for Collaborative Filtering (CF) demonstrate remarkable performance for one-class (implicit negative) recommendation tasks by extending autoencoders with relaxed but tractable latent distributions. Explicitly modeling a latent distribution over user preferences allows VAEs to learn user and item representations that not only reproduce observed interactions, but also generalize them by leveraging learning from similar users and items. Unfortunately, VAE-CF can exhibit suboptimal learning properties; e.g., VAE-CFs will increase their prediction confidence as they receive more preferences per user, even when those preferences may vary widely and create ambiguity in the user representation. To address this issue, we propose a novel Queryable Variational Autoencoder (Q-VAE) variant of the VAE that explicitly models arbitrary conditional relationships between observations. The proposed model appropriately increases uncertainty (rather than reduces it) in cases where a large number of user preferences may lead to an ambiguous user representation. Our experiments on two benchmark datasets show that the Q-VAE generally performs comparably or outperforms VAE-based recommenders as well as other state-of-the-art approaches and is generally competitive across the user preference density spectrum, where other methods peak for certain preference density levels.

Keywords: One-Class Collaborative Filtering; Variational Autoencoder; Conditional Inference.

ACM Reference Format:
Ga Wu, Mohamed Reda Bouadjenek, and Scott Sanner. 2019. One-Class Collaborative Filtering with the Queryable Variational Autoencoder. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '19), July 21–25, 2019, Paris, France.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3331184.3331292

1 INTRODUCTION

Autoencoder-based Collaborative Filtering (CF) algorithms make predictions by embedding user preferences into a latent space that enables generalization to unobserved user preferences [1]. However, a conventional Autoencoder recommender tends to be unsatisfactory as latent representations are likely to overfit and memorize individual observations [2]. Indeed, an Autoencoder-based model for CF may be overly sensitive to individual user-item interactions, and thus may significantly change the latent representation of a user even with a single interaction update. Several prior works have noted this unsatisfactory representation issue [2, 3] and thus Denoising Autoencoders [4] have been developed to mitigate this issue. Unfortunately, denoising can hurt the prediction performance when the data is very sparse as we show later in the experiments.

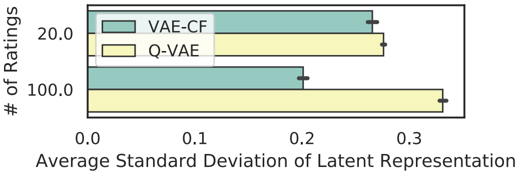

Figure 1: In this experiment, we show the average standard deviation of the diagonal Gaussian latent embeddings for VAE-CF and Q-VAE across 500 users. At the top, we first measure this embedding uncertainty after sampling 20 real interactions from each user's data and at the bottom we add in 80 random (fake) interactions. While Q-VAE increases its uncertainty, VAE-CF oddly becomes more certain in user preferences after observing this incoherent random data.

Recently, Variational Autoencoders (VAEs) [5] – which model *distributions* over latent representations – have been used and extended by Liang et al. [6] for CF recommendation (VAE-CF) and showed remarkable prediction performance improvement over previous Autoencoding methods. The prediction performance improvement arising from the generalization of VAEs over non-probabilistic Autoencoders is due to two key reasons: (i) VAEs relax the latent distribution from a (deterministic) Delta function to a Gaussian distribution allowing for explicit representation of user and item uncertainty, and (ii) VAEs regularize the latent distribution through Kullback-Leibler (KL) divergence with a tractable standard Gaussian distribution leading to learning stability (i.e., less sensitivity to individual data). Despite their remarkable prediction performance, VAEs exhibit the undesirable property of being over-confident when users express a large number of preferences. We argue in this paper that this property is particularly problematic because VAE-CF tends to recommend items to users with high certainty if a user has a considerable number of observed interactions even when these preferences may vary widely and increase ambiguity in the latent user representation as demonstrated in Figure 1.

To address the issue mentioned above, we propose the Queryable Variational Auto-encoder (Q-VAE) for one-class (implicit negative)

[*] Affiliate to Vector Institute of Artificial Intelligence, Toronto

SIGIR '19, July 21–25, 2019, Paris, France
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6172-9/19/07…$15.00
https://doi.org/10.1145/3331184.3331292

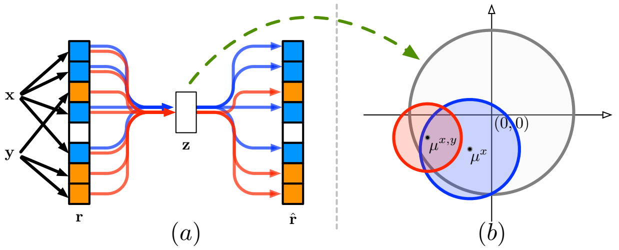

Figure 2: Proposed Q-VAE model. (a) The joint likelihood $\log p(\mathbf{x})$ and conditional likelihood $\log p(\mathbf{y}|\mathbf{x})$ objectives share the VAE network parameters that form a structured regularization. (b) $KL[q(\mathbf{z}|\mathbf{x},\mathbf{y})||q(\mathbf{z}|\mathbf{x})]$ restricts the user representation from severe changes with additional observations $\mathbf{y}$.

recommendation tasks. The key contribution of the Q-VAE is to reformulate the variational lower-bound of the joint observation distribution to support arbitrary conditional queries over *observed* user interactions. We show that our model can accurately measure the uncertainty of user latent representations (cf. Figure 1), thus preventing the model from performing poorly for users with a large number of interactions. Finally, we empirically demonstrate that Q-VAE outperforms VAE-CF in terms of prediction performance and is also competitive w.r.t. several state-of-the-art recommendation algorithms across the user preference density spectrum.

2 QUERYABLE-VAE FOR RECOMMENDATION

We begin with notation: we denote the observed preferences of user i as a set $\mathbf{r}_i$ of items preferred by the user (assuming binary preference with only positive observations for the one-class case). We denote partial preference observation subsets as $\mathbf{x}_i$ and $\mathbf{y}_i$, where $\mathbf{x}_i, \mathbf{y}_i \subseteq \mathbf{r}_i$, $\mathbf{x}_i \cap \mathbf{y}_i = \emptyset$, and $\mathbf{x}_i \cup \mathbf{y}_i \subseteq \mathbf{r}_i$. In the following context, we omit the user subscript i to reduce notational clutter.

We propose the Queryable Variational Auto-encoder (Q-VAE) to model the joint probability of a user's preferences $p(\mathbf{r})$ and the conditional probability $p(\mathbf{y}|\mathbf{x})$ of some subset of preferences given the others, which allows the model to treat the recommendation as a conditional inference (i.e., a *query*) problem with an arbitrary evidence set of user preference observations.

Instead of directly modeling the lower-bound of the log joint probability $p(\mathbf{r})$ as other VAE-based recommender systems do, we propose to model the joint probability of any arbitrary partition of $\mathbf{x}$ and $\mathbf{y}$ as follows:

$$\log p(\mathbf{x},\mathbf{y}) = \log p(\mathbf{y}|\mathbf{x}) + \log p(\mathbf{x}) \tag{1}$$

where $\log p(\mathbf{y}|\mathbf{x})$ estimates user preference for some items given the user's historical interactions, and $\log p(\mathbf{x})$ estimates how well the model can reproduce the historical interactions.

Maximizing both $\log p(\mathbf{y}|\mathbf{x})$ and $\log p(\mathbf{x})$ for a given user is intractable due to the unknown relations between their interactions. We therefore optimize the lower-bounds $\log p(\mathbf{x})$, which has been derived for VAE [6–8] as follows:

$$\log p(\mathbf{x}) \geq E_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - KL[q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})] \tag{2}$$

where ϕ and θ are respectively encoder and decoder coefficients, and $\mathbf{z}$ is a user latent representation. Similarly, we can define the lower-bound of $\log p(\mathbf{y}|\mathbf{x})$ as follows:

$$\log p(\mathbf{y}|\mathbf{x}) \geq E_{q(\mathbf{z}|\mathbf{x},\mathbf{y})}[\log p(\mathbf{y}|\mathbf{z})] - KL[q(z|\mathbf{x},\mathbf{y})||p(\mathbf{z}|\mathbf{x})] \tag{3}$$

However, we note that we cannot form Equation 3 into an Autoencoder since the distribution $p(\mathbf{z}|\mathbf{x})$ is unknown. While it is possible to relax $p(\mathbf{z}|\mathbf{x})$ by $p(\mathbf{z})$ as it has been done in both CVAE [4] and BCDE [9], in this work, we require a variational approximation $q(z|\mathbf{x},\mathbf{y})$ with additional observations $\mathbf{y}$ as close as possible to $p(\mathbf{z}|\mathbf{x})$ to ensure that recommendations align with observed preferences $\mathbf{x}$.

We address this excessive relaxation problem by approximating the prior distribution $p(z|x)$ with its lower-bound $q_\phi(\mathbf{z}|\mathbf{x})$ learned from Equation 2. Thus, Equation 3 can represent a second VAE objective function as follows:

$$\log p(\mathbf{y}|\mathbf{x}) \geq E_{q_\psi(\mathbf{z}|\mathbf{x},\mathbf{y})}[\log p_\vartheta(\mathbf{y}|\mathbf{z})] - KL[q_\psi(z|\mathbf{x},\mathbf{y})||q_\phi(\mathbf{z}|\mathbf{x})] \tag{4}$$

where ψ and ϑ are respectively encoder and decoder coefficients of the second VAE. The naive combination of the two VAE objectives from Equations 2 and 4 is impractical due to the need to maintain two VAE parameter sets and obtain the conditional prior $q_\theta(\mathbf{z}|\mathbf{x})$ before training the second VAE network.

We mitigate the above problem by sharing the parameter sets from the two VAE objectives and training the two networks simultaneously. Specifically, Q-VAE optimizes a combined objective function on a single VAE network structure as follows:

$$\begin{aligned} &\log p(\mathbf{x},\mathbf{y}) \\ &\geq E_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})dz] - KL[q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})] \\ &+ E_{q_\phi(\mathbf{z}|\mathbf{x},\mathbf{y})}[\log p_\theta(\mathbf{y}|\mathbf{z})dz] - KL[q_\phi(z|\mathbf{x},\mathbf{y})||q_\phi(\mathbf{z}|\mathbf{x})], \end{aligned} \tag{5}$$

where the two sub-objective functions form a mutually structured regularizer as demonstrated in Figure 2(a).

Arbitrary Combination of Evidence and Query Variables: Instead of fixing the split of variables $\mathbf{x}$ and $\mathbf{y}$ as in CVAE and BCDE, Q-VAE randomly splits variables during training through a dropout method as shown in Equation 6 that is detailed later. Such random dropout training enables the model to do arbitrary conditional inference with a different set of evidence or query variables without retraining a new model. Specifically, we can obtain random $\mathbf{x} \cup \mathbf{y}$, $\mathbf{x}$ and $\mathbf{y}$ as follows:

$$\mathbf{x}\cup\mathbf{y} = Dropout(\mathbf{r}, \rho); \quad \mathbf{x} = Dropout(\mathbf{x}\cup\mathbf{y}, \rho); \quad \mathbf{y} = \mathbf{x}\cup\mathbf{y} - \mathbf{x}, \tag{6}$$

where ρ indicates the dropout ratio that randomly samples dropout percentages uniformly from the range $[0.1, 0.5]$.

KL Divergence: The objective function in Equation 5 introduces one additional KL divergence term that regularizes posterior distributions $q_\phi(\mathbf{z}|\mathbf{x})$ and $q_\phi(z|\mathbf{x},\mathbf{y})$. Since both posterior distributions are approximated as diagonal Gaussian distributions that are parameterized by mean $\boldsymbol{\mu}$ and standard derivation $\boldsymbol{\sigma}$, the KL divergence computation is in closed-form and is computed as follows:

$$KL[q(\mathbf{z}|\mathbf{x},\mathbf{y})||q(\mathbf{z}|\mathbf{x})] = \sum_k \left[\log \frac{\sigma_k^x}{\sigma_k^{x,y}} + \frac{(\sigma_k^{x,y})^2 + (\mu_k^{x,y} - \mu_k^x)^2}{2(\sigma_k^x)^2} - \frac{1}{2}\right] \tag{7}$$

where k is the index of latent dimension. The KL divergence suggests to keep expectation of the user preference $\boldsymbol{\mu}^{x,y}$ as close as possible to $\boldsymbol{\mu}^x$ when the model observes more interactions $\mathbf{y}$ as demonstrated in Figure 2(b).

Table 1: Results of Movielens-1M dataset with 95% confidence interval. Hyper-parameters are chosen from the validation set. α: loss-weighting. λ: L2-regularization. ρ: corruption rate.

model	rank	α	λ	epochs	ρ	R-Precision	MAP@5	MAP@50	Precision@5	Precision@50	Recall@5	Recall@50
PureSVD	50	0	1	10	0	0.092±0.0024	0.1212±0.0052	0.0987±0.0024	0.116±0.0043	0.0852±0.0018	0.0383±0.002	0.2383±0.0052
BPR	200	0	1e-5	30	0	0.0933±0.0025	0.1192±0.0052	0.1002±0.0025	0.1141±0.0043	0.0875±0.0019	0.0375±0.002	0.2426±0.0052
WRMF	200	10	100	10	0	0.097±0.0026	0.1235±0.0053	0.1039±0.0025	0.1198±0.0045	0.091±0.002	**0.0411±0.0022**	**0.2668±0.0058**
CDAE	200	0	1e-5	300	0.5	0.0941±0.0025	0.1297±0.0056	0.1032±0.0028	0.1226±0.0047	0.0891±0.0021	0.035±0.0018	0.2177±0.0047
VAE-CF	200	0	1e-5	200	0.4	0.0892±0.0025	0.1066±0.0048	0.093±0.0022	0.1054±0.0039	0.0827±0.0017	0.0376±0.002	0.2449±0.0054
AutoRec	200	0	1e-5	300	0	0.0945±0.0025	0.1254±0.0054	0.1017±0.0026	0.1194±0.0045	0.0877±0.0019	0.0377±0.002	0.2398±0.0052
Q-VAE	200	0	0.1	200	0	**0.1±0.0026**	**0.1306±0.0055**	**0.1066±0.0026**	**0.125±0.0046**	**0.0917±0.0020**	0.0404±0.0021	0.2504±0.0054

Table 2: Results of Netflix dataset with 95% confidence interval. Hyper-parameters are chosen from the validation set.

model	rank	α	λ	epochs	ρ	R-Precision	MAP@5	MAP@50	Precision@5	Precision@50	Recall@5	Recall@50
PureSVD	50	0	1	10	0	0.0994±0.0003	0.159±0.0007	0.118±0.0003	0.146±0.0005	0.0953±0.0003	0.0445±0.0003	0.2188±0.0006
BPR	50	0	1e-5	30	0	0.0757±0.0002	0.1197±0.0006	0.096±0.0003	0.115±0.0005	0.0816±0.0002	0.0291±0.0002	0.1859±0.0006
WRMF	200	10	1e4	10	0	0.0985±0.0003	0.1531±0.0007	0.117±0.0003	0.1447±0.0006	0.096±0.0003	0.045±0.0003	**0.2325±0.0007**
CDAE	50	0	1e-5	300	0.2	0.0797±0.0003	0.1251±0.0006	0.0979±0.0003	0.1198±0.0005	0.0832±0.0002	0.0323±0.0002	0.1788±0.0006
VAE-CF	100	0	1e-4	300	0.5	**0.1017±0.0003**	0.1559±0.0007	0.1176±0.0003	0.1465±0.0005	0.0957±0.0003	**0.0467±0.0003**	0.2309±0.0006
AutoRec	50	0	1e-5	300	0	0.0876±0.0003	0.14±0.0006	0.1074±0.0003	0.1324±0.0005	0.0894±0.0003	0.0361±0.0002	0.1958±0.0006
Q-VAE	100	0	1e-5	200	0	0.0976±0.0003	**0.1593±0.0007**	**0.1194±0.0003**	**0.1488±0.0006**	**0.0972±0.0003**	0.0429±0.0003	0.2303±0.0006

Table 3: Summary of datasets used in evaluation.

Dataset	#Users	#Items	$\|r_{i,j} > \vartheta\|$	Sparsity
MovieLens-1m	6,038	3,533	575,281	2.69×10^{-2}
Netflix Prize	2,649,430	17,771	56,919,190	1.2×10^{-3}

3 EXPERIMENTS AND EVALUATION

In this section, we evaluate Q-VAE by comparing it to a variety of scalable state-of-the-art One-Class Collaborative Filtering (OC-CF) algorithms on two different benchmark datasets. The comparison includes: (i) recommendation precision performance, (ii) latent representation uncertainty evaluation, and (iii) convergence speed.

Datasets: We evaluate the candidate algorithms on two publicly available datasets: Movielens-1M[1] and Netflix Prize[2] where in both datasets, ratings rages from 1 to 5. For both datasets, we binarize the ratings based on a threshold $\vartheta = 3$, defined to be the upper half of the range of the ratings. Hence, a rating $r_{ij} > \vartheta$ is considered as a positive feedback, otherwise, it's considered as a negative feedback.

Evaluation Metrics: We evaluate the recommendation performance using five different metrics: Precision@K, Recall@K, MAP@K, R-Precision, and B-NDCG.

Candidate Methods: We compare Q-VAE with six different CF algorithms, ranging from classical Matrix Mactorization to the latest VAE for CF. These algorithms are:

- **PureSVD [10]:** A method that constructs a similarity matrix through randomized SVD decomposition of implicit matrix R.
- **BPR [11]:** Bayesian Personalized Ranking. One of the first recommender that explicitly optimize the pairwise ranking.
- **WRMF [12]:** Weighted Regularized Matrix Factorization. A Matrix Factorization that was designed for OC-CF.
- **AutoRec [1]:** Autoencoder based recommendation system with one hidden layer, Relu activation, and sigmoid cross entroy loss.
- **CDAE [4]:** Collaborative Denoising Autoencoder, which is specifically optimized for implicit feedback recommendation tasks.
- **VAE-CF [6]:** Variational Autoencoder for CF. A state-of-the-art metric learning based recommender system.

Ranking Performance Evaluation: Tables 1 and 2 show the general performance comparison of Q-VAE with the six baselines using R-Precision, MAP, Precision@K, and Recall@K metrics. From the results obtained, we make the following observations: (i) In general, Q-VAE achieves competitive prediction performance w.r.t. the state-of-the-art recommendation algorithms such as WRMF and VAE-CF. (ii) Also, Q-VAE outperforms all candidates on MAP and Precision@K, at the expense of Recall@K as a trade-off.

Performance vs. User Interaction Level: We investigate conditions under which Q-VAE achieves a significantly higher prediction performance than the baselines. To this end, we categorize users based on their number of interactions in the training set into 4 categories. The categories come from the 25%, 50%, 75% quartiles of the number of training interactions, which indicate how often the user rated items in the training set.

Figure 3 shows the performance comparison for different user categories. We note that CDAE, comparing to AutoRec, performs poorly for users with sparse historical interactions. It reflects our intuition that simple random corruption of inputs (i.e., "Denoising") hurts the performance for users with sparse observations. WRMF and VAE-CF both perform well with sparse user interactions but poorly for users with many interactions. In comparison, Q-VAE shows relatively stable and good performance over all four user categories and significant prediction performance improvement over VAE-CF, especially with a large number of user interactions.

User Representation Uncertainty: Both VAE-CF and Q-VAE explicitly model the user latent representation distributions. Hence, in

[1] https://grouplens.org/datasets/movielens/1m/

[2] https://www.kaggle.com/netflix-inc/netflix-prize-data

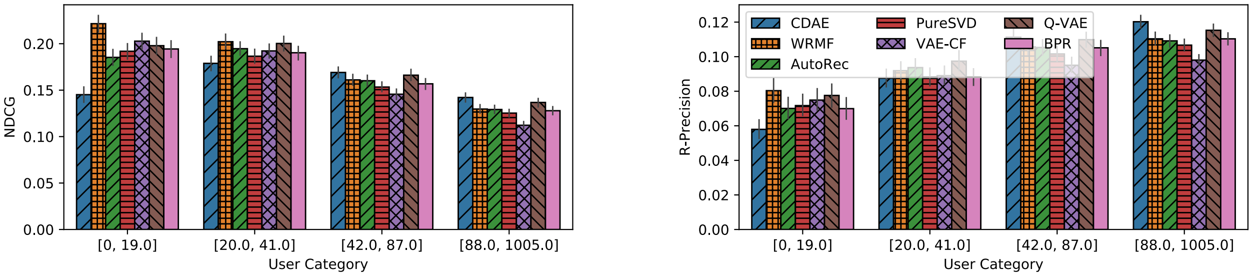

Figure 3: Average NDCG comparison for different quantiles of user activity (number of ratings as binned into the ranges shown in $[\cdot, \cdot]$) for MovieLens-1m. Error bars show the standard deviation of the NDCG across users in that bin.

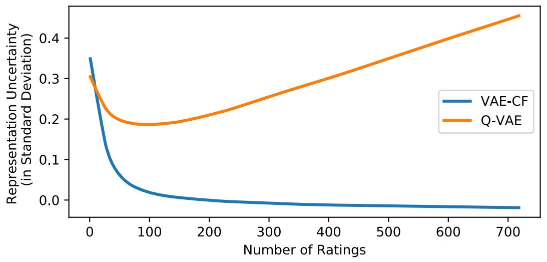

Figure 4: Average standard deviation of the diagonal Gaussian latent representations of users (averaged over users having a given number of ratings) for VAE-CF and Q-VAE on Movielens-1m; VAE-CF is overconfident with a high number of user ratings while Q-VAE shows more uncertainty.

this experiment, we analyze the latent representation uncertainty of users vs. their number of ratings. As shown in Figure 4, VAE-CF tends to provide high certainty to users with a large number of interactions, even though a large number of interactions often requires more uncertainty to cover the range of preferences. The caveat of this over-certainty is reflected in our observation of Figure 3, where VAE-CF performs poorly for users with a large number of interactions as compared to Q-VAE (rightmost chart).

Convergence Profile: We track the convergence progress among the four Autoencoder based recommenders in Figure 5. It shows that VAE-based algorithms converge faster than the original Autoencoder approaches (which tend to overfit). Q-VAE benefits from relatively fast and smooth convergence without overfitting due to the mutually structured regularization of its two objectives.

4 CONCLUSION

In this paper, we proposed the Queryable Variational Auto-encoder (Q-VAE) as a way to explicitly condition recommendations in one-class collaborative filtering on observed user preferences to better model latent uncertainty of user preferences. Our experiments show that the Q-VAE not only converges faster, but also outperforms several state-of-the-art Auto-encoder based recommendation models. Also, we showed that Q-VAE avoids over-confidence with a large number of user preferences leading to strong recommendation performance across the user preference density spectrum.

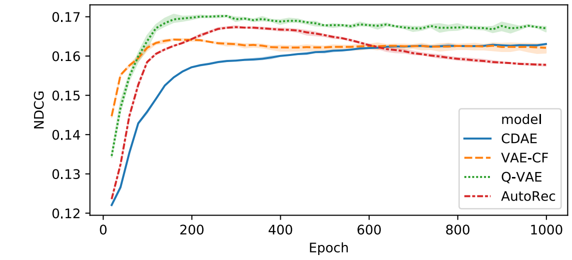

Figure 5: NDCG versus training epochs for the four Autoencoder based recommendation algorithms.

REFERENCES

[1] Suvash Sedhain, Aditya Krishna Menon, Scott Sanner, and Lexing Xie. Autorec: Autoencoders meet collaborative filtering. In *Proceedings of the 24th International Conference on World Wide Web*, pages 111–112. ACM, 2015.

[2] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(Dec):3371–3408, 2010.

[3] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103. ACM, 2008.

[4] Yao Wu, Christopher DuBois, Alice X Zheng, and Martin Ester. Collaborative denoising auto-encoders for top-n recommender systems. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, pages 153–162. ACM, 2016.

[5] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. 2013.

[6] Dawen Liang, Rahul G Krishnan, Matthew D Hoffman, and Tony Jebara. Variational autoencoders for collaborative filtering. *arXiv preprint arXiv:1802.05814*, 2018.

[7] Xiaopeng Li and James She. Collaborative variational autoencoder for recommender systems. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 305–314. ACM, 2017.

[8] Yifan Chen and Maarten de Rijke. A collective variational autoencoder for top-n recommendation with side information. In *Proceedings of the 3rd Workshop on Deep Learning for Recommender Systems*, pages 3–9. ACM, 2018.

[9] Rui Shu, Hung H. Bui, and Mohammad Ghavamzadeh. Bottleneck conditional density estimation. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pages 3164–3172. JMLR.org, 2017.

[10] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 39–46. ACM, 2010.

[11] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*, pages 452–461. AUAI Press, 2009.

[12] Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, pages 263–272. Ieee, 2008.