ICIC Express Letters
Volume **17**, Number **2**, February **2023**

pp. **163–170**

BILATERAL VARIATIONAL AUTOENCODER OPTIMIZATION WITH LEARNING-TO-RANK METRIC ON COLLABORATIVE FILTERING RECOMMENDATION

Giovinno Bryan William[1] and Alexander Agung Santoso Gunawan[2,*]

[1]Computer Science Department, BINUS Graduate Program – Master of Computer Science
[2]Computer Science Department, School of Computer Science
Bina Nusantara University
Jl. K. H. Syahdan No. 9, Kemanggisan, Palmerah, Jakarta 11480, Indonesia
giovinno.william@binus.ac.id; *Corresponding author: aagung@binus.edu

Received February 2022; accepted May 2022

Abstract. *Common method in the development of recommendation models involves the concept of Collaborative Filtering (CF). Traditional CF models such as matrix factorization will create a linear relation of user and item latent vector from the sparse user-item interaction matrix as input for rating prediction. The limited linear representation of latent vectors in matrix factorization leads to experimentations of Variational Autoencoders (VAE) in CF to evaluate nonlinear relations of latent vectors while also providing a more effective prediction mechanism on matrix factorization problems. This paper proposed a deeper implementation of VAE in CF by combining architecture of Bilateral Variational Autoencoder (BiVAE) and Ranking-Critical Training (RaCT). BiVAE splits the inference model of VAE into user and item inference model in order to solve dyadic data in the interaction matrix. The performance of BiVAE is evaluated using RaCT, an actor-critic neural-network architecture which shifts the multinomial likelihood to ranking optimization as the evaluation objective. Experiments are conducted by comparing the performance of the proposed architecture with RaCT and BiVAE using Normalized Discounted Cumulative Gain (NDCG) and Recall metrics. Experimental results showed improvement in performance metric, outperforming baseline models with average margin of 0.03 for Recall and 0.02 for NDCG across all datasets.*
Keywords: Variational autoencoder, Collaborative filtering, Learning-to-rank, Recommender system, Actor-critic

1. **Introduction.** The creation of a recommendation system is built upon the goal of achieving a system that can understand the preferences of users. Common techniques are implemented to accommodate this issue including Collaborative Filtering (CF), Content-Based Filtering (CBF), or hybrid filtering. Among these methods, CF has received the most attention. This is because users are more interested in products that are already liked or that are given a higher preference by other users who have similar interests [1]. In addition, a literature review showed the trends of research on the CF method were higher than other methods [2].

There is one major problem that hinders the CF technique that other techniques do not face, namely data sparsity. In its implementation, the recommendation system uses large amounts of data in the form of an interaction matrix between the user and the item. In reality, a user is unlikely to interact with all items and tends to interact with a small part of the set of items. This results in the interaction matrix containing empty values, referring to users and items that have never interacted with each other. The existence of large data in which most of the values are empty can be referred to as data sparsity.

DOI: 10.24507/icicel.17.02.163

The predictions made by the recommendation system will be more accurate if the trained data has a low level of sparsity.

Several studies have tried to lighten the burden of recommendation systems in dealing with data sparsity problems. Recent approaches have shown that using a Variational Autoencoder (VAE) on CF can give better results than some state-of-the-art models. The basis for using VAE is to generate/reconstruct generative data based on observational data. Using this concept, VAE is implemented in CF by training the model to observe a sparse user-item matrix and reconstructing the matrix into a dense matrix.

Two discrepancies are founded in the implementation of VAE models for CF. The main problem concerns the user-item dyadic data representation on a single vector VAE, while the other problem is in the VAE learning method which tries to maximize the likelihood of observation, contradicting to recommendation system metric that tries to increase the learning-to-rank objective. The Bilateral Variational Autoencoder (BiVAE) model was developed to position VAE as a model that accepts dyadic data [3], while the Ranking-Critical Training (RaCT) model tries to implement the actor-critical concept in reinforcement learning to change the model's objectives. VAE becomes learning-to-rank [4].

We want to conduct an experiment involving the combination of the two algorithms mentioned above with the main aim of easing the burden of the recommendation system (especially CF) in overcoming the problem of data sparsity. By adjusting BiVAE's learning metrics using the RaCT model in an actor-critic paradigm, implementing BiVAE as the actor model and RaCT as the critic model, we aim to create a model which can surpass the state-of-the-art benchmark on collaborative filtering use cases. This study consists of

a) Related works section, explaining previous studies related to the discovery of our proposed model;
b) Methodology section, explaining the concept behind the proposed model, the experimental design, the dataset to be used for the experiment, and its evaluation method;
c) Result and discussions section, explaining the results we achieve after conducting the experiment and the discovery we acknowledge during the analysis of our proposed model;
d) Conclusion section, which summarizes results achieved on this study and further improvements.

2. **Related Works.** The VAE concept was first implemented in CF through research by Li and She [5]. This study uses a Bayesian generative model called CVAE, where the model uses rating and content data to study unsupervised deep latent representation and to study the implicit relationship between users and items through content and ratings. This model applies the concept of Maximum a Posteriori estimation (MAP) and efficient variational inference using stochastic gradient variational Bayes.

Kuchaiev and Ginsburg demonstrated how highly complex autoencoders can be successfully trained even on relatively small amounts of data using a relatively new and structured (dropout) deep learning technique ("exponentially scaled linear units") [6]. They introduced iterative output refeeding, which is a technique that allows them to make solid updates in collaborative filtering, increase the speed of learning, and further improve the generalization performance of their models.

A VAE-CF model is proposed in the form of implementing a deep generative model on CF through a VAE form by emphasizing the adoption of the negative sample sampling concept to overcome the sparsity problem in the dataset with implicit feedback [7]. Several variations of the proposed model include VAE-CF which models the distribution of user responses, Conditional VAE (CVAE-CF) which models the conditional distribution of user responses when given supporting information, Joint Multi-modal VAE (JMVAE-CF) which models the combined distribution.

Several studies also implemented the use of VAE in collaborative filtering [8-11] with similar structures. Most of these models emphasize their enhancement by adjusting the probability function and enabling the use of side information in the form of user and item explicit features. Although these studies have proven the vast possibility of leveraging VAE for CF, we realize that none of these studies attempted to reconstruct the VAE models to further match the use case and objectives in recommendation system, which is to extract non-linear, two-way interaction data (or dyadic data) and to repurpose the learning objective to better suit the ranking mechanism in giving recommendations.

The learning objective in this study is taken from the research of Sam Lobel, who proposed a model called Ranking-Critical Training (RaCT) studying a distinguishable approach to ranking metrics, i.e., a predictive network, then leveraged as a target for optimization via gradient ascent [4]. This goes in contrast to existing methods of collaborative filtering, which defines an objective relaxation ahead of time. This learning methodology approaches non-optimized functions directly derived from the reinforcement learning actor-critical paradigm, adapted for collaborative screening.

The model BiVAE which will be later used as a base model in this study was proposed by Truong et al. [3]. They introduced a new hierarchical generative model of the dyad (user-item interaction) and with it a user-item inference model, which is parameterized using a neural network.

Based on the two models mentioned above, this study focused on implementing the architecture of both models by creating a joined model architecture which is explained in Section 3. This model will shift the generative purpose of VAE models into a discriminative model used for ranking predictions.

3. Methodology.

3.1. Proposed model.

In general, the input component of the recommendation system in the form of preference data is described as dyadic data. The initial design of VAE was intended for vector-based data, so that substantially it could not describe the bidirectional nature of dyadic data. Truong et al. modified VAE by changing the structure of the model and proposing a Bilateral Variational Encoder (BiVAE) [3]. There is a generative model that produces dyadic data in the form of user-item interactions, as well as two inference models (separately between users and items). The separation of the inference model allows the model to capture patterns of uncertainty that exist in users and items, resulting in a better model in dealing with sparse preference data than the usual VAE method.

Lobel and his team formulated an architecture to change the maximum likelihood objective so that the VAE model can be used for learning-to-rank objectives, where they proposed the Ranking-Critical Training (RaCT) algorithm [4]. The RaCT algorithm utilizes the concept of actors and criticism in reinforcement learning with a VAE-CF as an actor and an additional network that acts as a critique of the results given by actors.

This study will implement BiVAE as an actor and RaCT as the critic in solving the sparsity problem based on the learning-to-rank objective. Architecture of the proposed model is shown in Figure 1. The proposed architecture will be executed by remodelling the Evidence Lower Bound (ELBO) of BiVAE. Based on variational inference approach, a constant is needed to evaluate similarity of two distributions. *Kullback Liebler* (KL) *divergence* is used for this calculation. Training on VAE requires minimizing the value of KL divergence [12]. This can be done using the ELBO function written in Equation (1). On the BiVAE model, a separation of parametric computation is used to depict the dyadic data nature of user and item. This optimization is separated into the user parametric optimization block and item parametric optimization described in [3].

$$\boldsymbol{\mathcal{L}} = \sum_{u,i} \mathbb{E}_{q(\theta_u|r_{u*})}\mathbb{E}_{q(\beta_i|r_{*i})}[\log p(r_{ui}|\beta_u, \theta_i)] - \sum_{u} KL(q(\theta_u|r_{u*})||p(\theta_u))$$

$$- \sum_i KL(q(\beta_i|r_{*i})||p(\beta_i)) \quad (1)$$

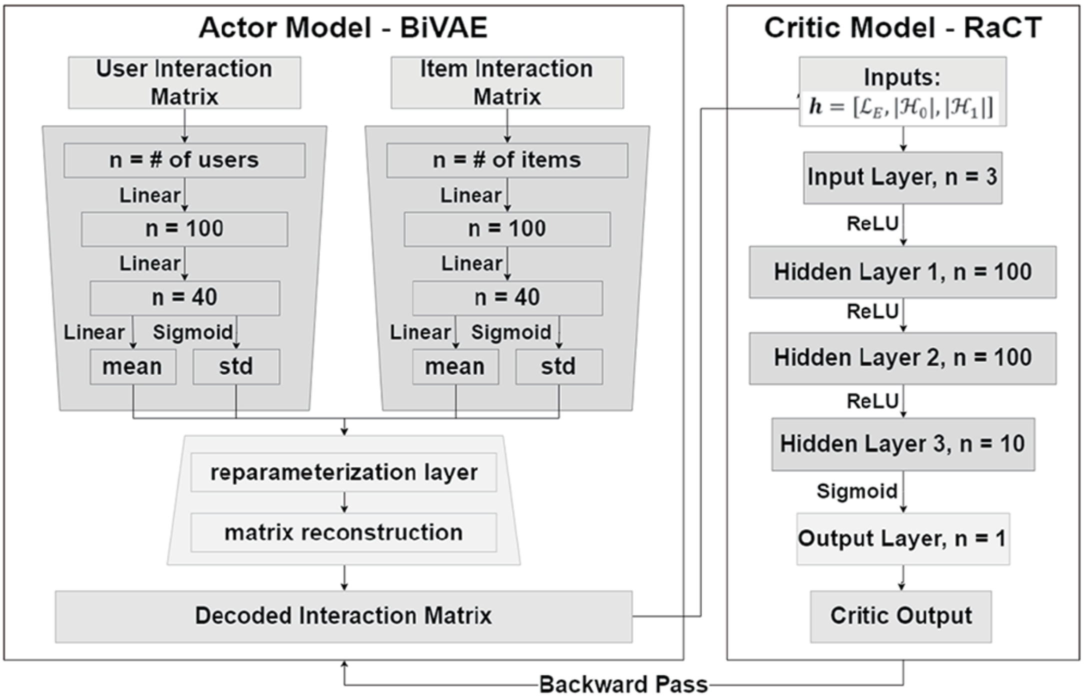


FIGURE 1. Proposed model architecture

To update the parametric values for decoder network p and encoder network q (φ-parameterized and τ-parameterized), we used RaCT as a critic model which is a π-parameterized neural network with input feature $\boldsymbol{h}$ shown in Figure 1, where $\mathcal{L}_E$ is the minimum likelihood value in the first part ELBO function, $|\mathcal{H}_0|$ is the number of unobserved items that a user will interact, and $|\mathcal{H}_1|$ is the number of observed items that user has interacted with. This neural network will be trained against mean-squared error metric as in Equation (2).

$$\boldsymbol{\mathcal{L}}_C(\boldsymbol{h}, y; \delta) = ||\boldsymbol{\omega}_\delta(\boldsymbol{h}) - y||^2 \quad (2)$$

BiVAE-RaCT model training will be conducted into several steps, which is actor pre-training, critic pre-training, and actor-critic training. To get a working baseline of likelihood value, the actor model will be trained until it reaches convergence. From that point, the critic model will use actor model's latest likelihood value as a feature to train itself. Once the critic model reaches convergence, it will update its own network parameters as well as the actor's network parameter. At the actor-critic training phase, each iteration will consist of one actor model forward pass, one critic model forward pass, and a connected backward pass computed using Equation (3).

$$\frac{\partial \boldsymbol{\mathcal{L}}_A}{\partial \pi} = \frac{\partial \boldsymbol{\mathcal{L}}_A}{\partial \boldsymbol{h}} \cdot \frac{\partial \boldsymbol{h}}{\partial \pi}; \quad \frac{\partial \boldsymbol{\mathcal{L}}_A}{\partial \varphi} = \frac{\partial \boldsymbol{\mathcal{L}}_A}{\partial \pi} \cdot \frac{\partial \pi}{\partial \varphi}; \quad \frac{\partial \boldsymbol{\mathcal{L}}_A}{\partial \tau} = \frac{\partial \boldsymbol{\mathcal{L}}_A}{\partial \pi} \cdot \frac{\partial \pi}{\partial \tau} \quad (3)$$

3.2. **Experimental design.** The code will be highly referenced from the study conducted in [3,4], while also being modified with additional libraries. Python libraries that will be used include common libraries such as numpy, pandas, matplotlib, and pytorch for building BiVAE and RaCT models. Meanwhile, the design of the model will utilize the Cornac library as a basic recommendation system prototyping library [13].

Hyperparameter tuning are conducted for the base BiVAE model to capture the optimum parameters used in this experiment. Based on those tuning, the number of latent dimensions K for each user and item representation is 20 for all models to be tested.

For autoencoder models, Multilayer Perceptron (MLP) will be used to find parameters in the inference model and generative model. Learning rate will be set to 0.0001, while the maximum number of epochs be set to 200 with early stopping conditions on 5 consecutive epochs. All models will be optimized using Adam, and batch size will be set to 128. For all datasets, a random split is done with distribution of 80% for training, 10% for validation, and 10% for testing.

After the model is completed, an automated process will be created that will test the proposed model along with several other baseline models against the evaluation method described in Section 3.3. The automation process is expected to produce a comparison of the evaluation values for each model being tested.

3.3. **Performance metric.** At the testing and evaluation stage, the model will predict the ranking by sorting the resulting multinomial probability values. For each user, a comparison will be made between the predicted film ranking and the original ranking of the film. The evaluation metric to be used consists of Recall@R and Normalized Discounted Cumulative Gain that has been deducted (NDCG@R) where the R value refers to the top R-item ranked by the system.

4. **Result and Discussion.**

4.1. **Data acquisition.** MovieLens [14] is a website that provides services in the form of a recommendation system and an online community for people who want to find movies. MovieLens dataset has been widely used to assist the research process, especially in the domain of recommendation systems. According to its publications, this dataset was downloaded more than 140,000 times in 2014 [15]. A search on Google Scholar shows more than 8,900 studies referring to the MovieLens dataset. There are different types of MovieLens datasets of different sizes. At the time this research was initiated, GroupLens accommodated various types of datasets ranging from MovieLens 100K (or often called small dataset), MovieLens 1M, MovieLens 10M, MovieLens 20M, MovieLens 25M, and MovieLens 1B. Each dataset is taken at a different time interval (larger interval distance for larger datasets). In this study, the datasets to be tested include MovieLens 10M, MovieLens 20M, and MovieLens 25M. Several previous studies explained that the MovieLens 100K and MovieLens 1M datasets are considered to be quite dense datasets so that the problem of data sparsity is not reflected in these datasets. On the other hand, MovieLens 1B is synthetic data, so its use in research is not recommended. A summary of the contents of the MovieLens dataset that will be used in this study can be seen in Table 1.

Table 1. Comparison of sparsity levels in the MovieLens 10M, 20M, and 25M datasets

Dataset	# users	# movies	# ratings	Sparsity
MovieLens 10M	71,567	10,681	10,000,054	98.69%
MovieLens 20M	138,493	27,278	20,000,263	99.47%
MovieLens 25M	162,541	62,423	25,000,095	99.75%

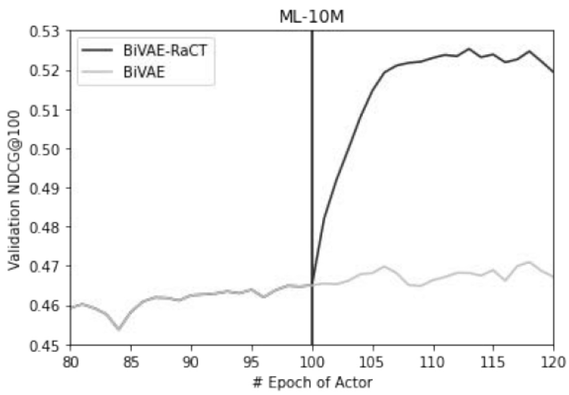
4.2. **Performance analysis.** This section reports the performance of our proposed method compared to different baselines across all datasets with NDCG and Recall as the performance metrics. The metrics are calculated using top-R item with R values of 50 for recall and 100 for NDCG. The different R values are based on the nature of corresponding metrics. Recall metric gives better overview of the model performance over small set of top-ranking items. On the contrary, NDCG metric depicts the overall performance gain of the model over the whole dataset, thus we used a bigger R value for NDCG. We also performed an ablation study towards selecting features suitable for the critic network.

Table 2 summarizes our tested model results. Our proposed method consistently outperforms the other comparative baselines by a significant margin. Across all datasets, our proposed method outperforms BiVAE with an average margin of 0.02 in NDCG@100 value and an average margin of 0.03 in Recall@50 value. This result provides positive evidence of how dyadic data can be handled with separate inference network and optimized using learning-to-rank objective functions. This opens the possibilities in BiVAE's constant adaptive priors with RaCT architecture to be applied on real world recommendation data which also relies on user and item attributes as implicit feedback.

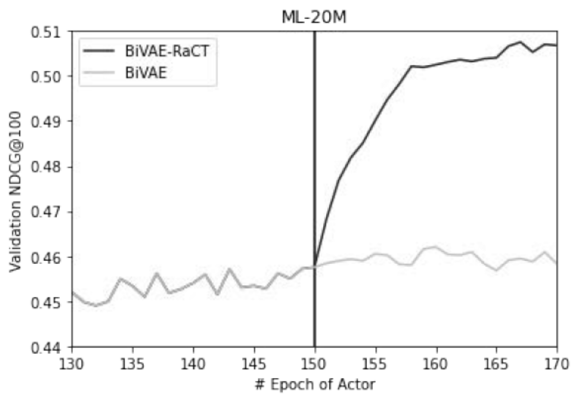
Analyzing the training and validation behaviour in Figure 2, we identify the effect of implementing critic model alongside a pre-trained actor model. Feedbacks given by the critic

Table 2. Quantitative performance result with Recall@50 and NDCG@100 metrics

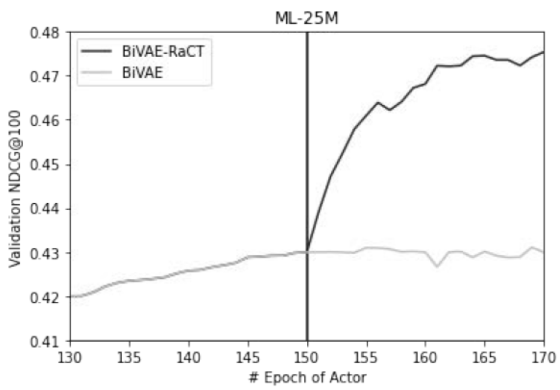
Models	MovieLens 10M		MovieLens 20M		MovieLens 25M	
	R@50	NDCG@100	R@50	NDCG@100	R@50	NDCG@100
BiVAE Bern	0.541	0.443	0.539	0.431	0.514	0.411
BiVAE Poiss	0.565	0.452	0.556	0.445	0.525	0.420
Mult-VAE (RaCT)	0.561	0.449	0.543	0.434	0.521	0.417
BiVAE-RaCT Bern (Own)	0.591	0.473	0.577	0.460	0.540	0.432
BiVAE-RaCT Poiss (Own)	**0.598**	**0.478**	**0.585**	**0.466**	**0.548**	**0.438**



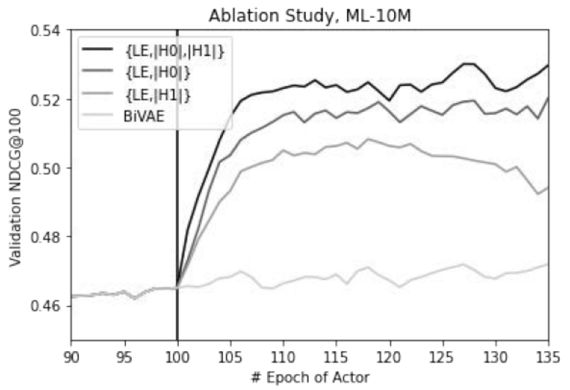
(a)



(b)



(c)



(d)

Figure 2. (a), (b), (c) Actor-critic model performance gain; (d) ablation study on critic model feature

model enable actor model to escape local optima and approach a higher global optimum. In other words, implementing the RaCT as a critic model allows BiVAE to obtain a higher convergence level. This behaviour is applied across all datasets. Thus, implementing RaCT on other collaborative models is possible, assuming the right feature is selected to train the critic model. Based on the dataset itself, we see an expected behaviour where smaller dataset requires less epoch to reach convergence point. MovieLens 20M and MovieLens 25M do not show any significant difference on epoch count since the sparsity value of mentioned dataset is nearly identical.

We conducted an ablation study upon pre-selected features for the critic model. We proposed three different features to be used in the model as mentioned in Section 3.1. Likelihood value is an essential factor on this critic model, since it creates the link for the critic model to backpropagate and create feedback for the BiVAE, we try the effects of appearance on the other two features. As shown in Figure 2(d), using all the features generates the best validation values. Removing $|\mathcal{H}_1|$ creates a slightly lower NDCG value, while removing the $|\mathcal{H}_0|$ reflects to a case of model overfitting. We conclude that the effects of each distinct feature in the critic model complements each other, in which removing any of the feature only results in a poorer model.

5. **Conclusion.** We presented a new architecture leveraging the previous work of BiVAE, a variational autoencoder model which is made specifically for dyadic data representation, where observations are broken down into measurements based on item and user object. The architecture is then optimized by changing the objective function for parametric adjustment in the inference models using RaCT, an architecture leveraging the concept of actor-critic paradigm. We provided theoretical base for the proposed architecture and deliberated the connections to previous works. Extensive experiments conducted to three MovieLens datasets show that the proposed architecture achieves significant improvements over other baseline VAE-CF methods, in which the proposed architecture outperforms other models by 0.02 in NDCG metric and 0.03 in Recall metric. Future work may be conducted on implementing BiVAE-RaCT architecture on real world dataset with the presence of user and item attributes as implicit interaction values.

REFERENCES

[1] M. Deshpande and G. Karypis, Item-based top-N recommendation algorithms, *ACM Transactions on Information Systems (TOIS)*, vol.22, pp.143-177, 2004.

[2] P. K. Singh, P. K. D. Pramanik, A. K. Dey and P. Choudbury, Recommender systems: An overview, research trends, and future directions, *International Journal of Business and Systems Research*, vol.15, no.1, pp.14-52, 2021.

[3] Q. T. Truong, A. Salah and H. W. Lauw, Bilateral variational autoencoder for collaborative filtering, *Proc. of the 14th ACM International Conference on Web Search and Data Mining*, pp.292-300, 2021.

[4] S. Lobel, C. Li, J. Gao and L. Carin, RaCT: Toward amortized ranking-critical training for collaborative filtering, *International Conference on Learning Representations (ICLR)*, 2020.

[5] X. Li and J. She, Collaborative variational autoencoder for recommender systems, *Proc. of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.305-314, 2017.

[6] O. Kuchaiev and B. Ginsburg, Training deep autoencoders for collaborative filtering, *arXiv.org*, arXiv: 1708.01715, 2017.

[7] W. Lee, K. Song and I.-C. Moon, Augmented variational autoencoders for collaborative filtering with auxiliary information, *Proc. of the 2017 ACM on Conference on Information and Knowledge Management*, pp.1139-1148, 2017.

[8] N. Sachdeva, G. Manco, E. Ritacco and V. Pudi, Sequential variational autoencoders for collaborative filtering, *Proc. of the 12th ACM International Conference on Web Search and Data Mining*, pp.600-608, 2018.

[9] Y. Chen and M. de Rijke, A collective variational autoencoder for top-N recommendation with side information, *Proc. of the 3rd Workshop on Deep Learning for Recommender Systems*, pp.3-9, 2018.

[10] D. Kim and B. Suh, Enhancing VAEs for collaborative filtering: Flexible priors & gating mechanisms, *Proc. of the 13th ACM Conference on Recommender Systems*, pp.403-407, 2019.

[11] I. Shenbin, A. Alekseev, E. Tutubalina, V. Malykh and S. I. Nikolenko, RecVAE: A new variational autoencoder for top-N recommendations with implicit feedback, *Proc. of the 13th International Conference on Web Search and Data Mining*, pp.528-536, 2020.

[12] D. P. Kingma and M. Welling, Auto-encoding variational Bayes, *International Conference on Learning Representations (ICLR)*, 2014.

[13] A. Salah, Q. Truong and H. W. Lauw, Cornac: A comparative framework for multimodal recommender systems, *Journal of Machine Learning Research*, vol.21, pp.1-5, 2020.

[14] GroupLens, *MovieLens*, https://movielens.org/, Accessed on Dec 16, 2021.

[15] F. M. Harper and J. A. Konstan, The MovieLens datasets: History and context, *ACM Trans. Interact. Intell. Syst.*, vol.5, 2016.