IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 8, 2019, accepted April 20, 2019, date of current version May 7, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2913468

Collaborative Variational Deep Learning for Healthcare Recommendation

XIAOYI DENG[1] **AND FEIFEI HUANGFU**[2]
[1]Business School, Huaqiao University, Quanzhou 362021, China
[2]College of Foreign Languages, Huaqiao University, Quanzhou 362021, China

Corresponding author: Xiaoyi Deng (londonbell.deng@gmail.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 71401058, and in part by the Program for New Century Excellent Talents in Fujian Province University (NCETFJ).

ABSTRACT Healthcare recommender system (HRS) has shown the great potential of targeting medical experts or patients, and plays a key role in improving an individual's health by providing insightful recommendations. The HRSs generate recommendations based on a successful and widely applied method known as collaborative filtering (CF). Despite its success, the CF suffers from data sparsity and cold-start problem, which results in the poor quality of recommendations. In particular, it is a great challenge to seeking information relevant to patients' condition, and understanding the medical terms and relationships between them in HRSs. To address these problems, we design a novel collaborative variational deep learning model (CVDL) to exploit multi-sourced information for providing appropriate healthcare recommendations in primary care service. CVDL employs additional variational autoencoder (VAE) to learn deep latent representations for item contents (the description of primary care doctors) in latent space, instead of observation space through an inference network. Meanwhile, the CVDL extracts latent user (patient) features by incorporating user profile in a VAE neural network. Therefore, the CVDL can learn better implicit relationships between items and users from item content, user profile, and rating matrix. In addition, a Stochastic Gradient Variational Bayes (SGVB) approach is proposed to calculate the maximum posterior estimates for learning model parameters. The experiments conducted on three datasets have indicated that our method significantly outperforms the state-of-the-art hybrid CF methods.

INDEX TERMS Collaborative topic regression, variational autoencoder, healthcare recommender system, side information, implicit feedback.

I. INTRODUCTION

Primary care acts as the principal point of both daily and long-term care for patients within a healthcare system [1]. It can facilitate the delivery of equitable healthcare and satisfy over 80% health needs of an individual throughout his/her life. However, patients usually face the challenge of looking for the right primary care physicians (doctors) without appropriate healthcare recommendation mechanism. Meanwhile, most primary care providers (PCPs) often lack the capability of transforming their services to more patient-centered approaches, which means most PCPs are unable to provide the applicable doctor recommendation service for patients [2]. The gap between the rapidly changing institutional environment and increasing patient autonomy makes it more complicated to recommend the most suitable doctors to patients. The increasing need of matching patients and doctors results in the presence of healthcare recommender systems (HRS) [3].

HRS can help to target patients (users) or medical experts (items) based on their medical records, and has shown huge potential of improving an individual's health by providing insightful recommendations, such as medication, diagnosis, treatment and even primary care physicians. Collaborative filtering (CF) is one of the key techniques to build a patient-centered HRS, due to its accuracy and scalability [4]. The essence of CF is to infer users' preferences from the behavior data of themselves and other users, and CF only depends on user-item rating which indicate how much users liked items. Most traditional CF methods are based on matrix factorization (MF) [5], which maps users and items into a shared latent space and utilizes a latent feature vector to represent either a user or an item [6]. However, MF-based models suffer from data sparsity and cold-start problem, so that

The associate editor coordinating the review of this manuscript and approving it for publication was Sabah Mohammed.

the accuracy of learning latent user/item representations is limited. To address these problems, large numbers of previous works incorporate user-specific or item-specific information into traditional MF. In order to extract more accurate latent factors from auxiliary information of user/item, some studies employ latent Dirichlet allocation (LDA) [7], [8], Bayesian personalized ranking [9], [10], denoising autoencoder (DAE) and its variants [11]–[14] to model user/item side information.

Among those methods mentioned above, collaborative topic regression (CTR) [8] is a probabilistic graphical model, which seamlessly integrates the conventional MF model with probabilistic topic modeling, and can generate more accurate recommendations based on item contents and other user's ratings. To improve CTR, previous works [15], [16] integrate social matrix factorization (SMF), into CTR model for jointly taking advantage of user ratings, user contexts, item contents and social relationships to achieve better predicting performance. By contrast, some studies directly learn the attention for rating prediction, where users allocate to their neighbors without uniformity [17], or directly extend CTR by integrating the user rating, item content, and social ensemble among items into the same hierarchical Bayesian model [18], [19]. These enhanced models are simple in principle and follow the same approximate inference procedure in a batch learning mode, but their representation capability is limited by LDA, and latent representation learning is not effective enough as the side information is very sparse.

In the last decade, deep learning models, e.g. convolutional neural network (CNN), recurrent neural network (RNN) and autoencoder (AE) have been taken advantage of in many fields, such as industrial design [20] and image recognition [21]. Lately, some works have applied variational autoencoder (VAE) [22] to perform CF task in recommendation, such as CVAE [23], CAVAE [24], CLVAE [25] and VAECF [26]. VAE is a non-linear probabilistic model, and it has the capability of capturing non-linearity and uncertainty in recommender systems with big data. Despite the effectiveness of these VAE-based methods, there are still several drawbacks, such as, CVAE and CAVAE directly use content information to extracts the latent item vectors. CLVAE and VAECF only exploit rating data, which results in poor performance under extremely high data sparsity scenario and cannot deal with cold-start problem [27].

In primary care systems, the additional information of patients and doctors are very rich, and have not been fully utilized for the improvement of recommendation performance, which makes HRS still in their infancy concerning trustworthiness and reliability. To solve those problems above, we devise a collaborative variational deep learning model (CVDL) for HRS in primary care, to provide insight and personalization into the care of patients by using their preferences. CVDL generates both latent user/item vectors through a variational neural network framework, which can effectively learn non-linear latent representations of users and items for further CF task. Meanwhile, the side information of user and item is incorporated into their latent factors generative processes, which means CVDL can mitigate data sparsity and model better latent representations of users and items. In inference process, we derived a Stochastic Gradient Variational Bayes (SGVB) approach to infer the posterior of latent factors of users and items, which makes the parameters of our model can be effectively learned. The rest of our paper is arranged as follows: Section 2 provides an overview of related works on CTR models. Section 3 introduces the CDVL model, and discusses parameters learning process. Section 4 shows experimental results and discussions, followed by conclusions and future work in section 5.

II. RELATED WORK

CTR utilizes item content to enhance CF methods and has achieved promising performance by integrating both user rating and item content [8], as shown in Figure 1. CTR combines the merits of both probabilistic MF (PMF) and topic modeling (LDA) models, and includes the latent variable for offsetting the topic proportions when modeling the user ratings, and the offset variable can successfully capture the item preference for a particular user considering their ratings.

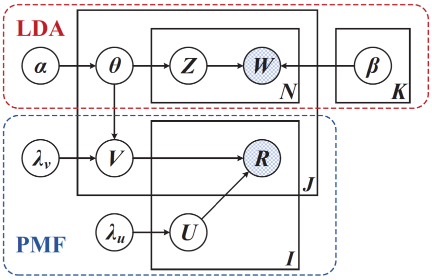

FIGURE 1. The framework of CTR.

However, CTR does not exploit user information and cannot learn reliable latent user representations. To address this issue, some studies have been proposed using different variants that incorporate social information into CTR. For instance, CTRSMF [15] and C-CTR-SMF2 [16] integrated CTR with SMF model using a strategy that is similar to SoRec, where the social relationships are simultaneously factorized with the rating matrix. However, they do not reveal the underlying relations among users due to the lack of physical explanation. Compared to CTRSMF and C-CTR-SMF2, LACTR [17] and RCTR [18] directly learn the amount of attention that users allocate to other users and leverages this learned influence to alleviate sparsity problem. These two methods assume that the social interactions of users usually follow topically similar contents, so they are very sensitive to different type of datasets and the prediction accuracy may vary with the distributions of datasets. For social recommendation, CTRSTE [19] integrates user ratings, item contents and trust ensemble into CTR, which is simple in algorithmic principle, but its representation capability is limited due to LDA model, and the latent representation learned is not effective enough when the side information is very sparse.

Recently, several works utilize deep learning models to help perform the CF task in CTR, due to the non-linear nature of neural networks, such as CDL [11], CVAE [23], CAVAE [24] and CTRDAE [28]. All of CDL, CVAE and CAVAE combine stacked DAE (SDAE) or VAE with CTR, and enable themselves to balance the influences of user ratings and item content, but the auxiliary information of user profile is not considered at al. By contrast, CTRDAE employs DAE and LDA to learn user social representation and item content representation respectively, to prevent user correlation overfitting under the sparse social relations scenarios. However, the content representation capability of CTRDAE is the same as CTR, which is limited due to topic modeling model. Although these works have improved CTR in separate aspects by using either content or social network information, a critical problem remains,*i.e.*, how to effectively integrate item contents, user ratings and user profiles/relations into CTR [29]. Unlike previous CTR-based recommendation methods, this paper constructs the generative processes of users and items through a neural variational framework, which enables our model to capture non-linear latent representations of both users and items.

III. COLLABORATIVE VARIATIONAL DEEP LEARNING FOR HRS

In this section, we introduce the collaborative variational deep learning model (CVDL) for HRS, which contains two main components: the deep generative model for feature extraction and PMF model for rating prediction, as shown in Figure 2.

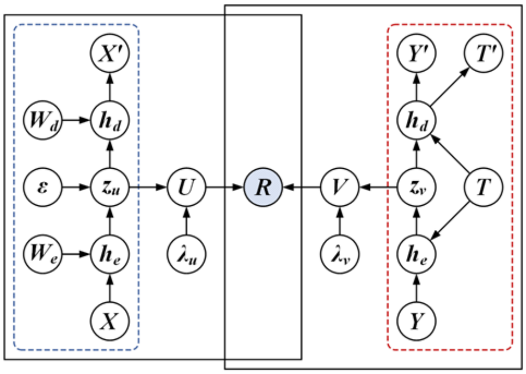

FIGURE 2. The graphic model of CVDL.

In CVDL, the items contents and user profiles/relations are generated by their latent variables, and rating predictions are generated jointly through both latent item and user variables. Latent item variables are incorporated with item contents information through latent content variables by employing an additional VAE model, due to the variety of. Latent user variables are linked with user profiles/relations via latent trust variables by a standard VAE model. Then, the users' and items' latent vectors are fed into the PMF model to learn the user-item relations, and finally predict the ratings.

A. NOTATIONS

Table 1 summarizes the symbols and notations in this paper. Given M users and N items, the latent factors of user and item are denoted by $U = \{u_i|i = 1, \ldots, M\} \in \mathbb{R}^{K\times M}$ and $V = \{v_j|j = 1, \ldots, N\}\in \mathbb{R}^{K\times N}$ respectively, where K denotes the dimensions of latent factors. For implicit feedback, the user rating matrix is denoted by $R \in \mathbb{R}^{M\times N}$, where $R_{ij} = 1$ indicates that the i-th user has interacted with the j-th item, otherwise $R_{ij} = 0$.

TABLE 1. Symbols and notations.

Symbols	Description
$R(R_{ij})$	Latent rating factors
$U(u_i)$	Latent user factors
$V(v_j)$	Latent item factors
$X(X_i)$	Latent user profile factors
$Y(Y_j)$	Latent item content factors
$T(T_{js})$	Latent tag factors of item content
W, b	Weight matrix and bias of VAE
h	Hidden layers of VAE
K	The dimension of latent space

The side information of users and items is denoted by two bag-of-words vectors over users and items, $X = \{X_i|i = 1, \ldots, M\} \in \mathbb{R}^{P\times M}$ and $Y = \{Y_j|j = 1, \ldots, N\} \in \mathbb{R}^{Q\times M}$, respectively, where P and Q are the dimensions of user side information and item side information respectively. Here, we call X and Y latent profile representation and latent content representation, respectively. Besides, the tag information of item contents are represents by $T\in \mathbb{R}^{N\times S}$, and T is a binary matrix, where $T_{js} = 1$ means s-th tag is associated with item v_j and $T_{js} = 0$ otherwise. Given R, X, Y and T, the problem is to infer latent user factor u_i and latent item factor v_j, and then to predict the unknown ratings R^*.

B. FEATURE EXTRACTION

As mentioned in [26], most MF-based models assume that the prior distributions of user and item latent factors are standard Gaussian distributions, and predict rating only through rating data. To extract more effective latent vectors from side information, CVDL incorporates both user's and item's side information into feature extraction, which can make positive contributions to the further rating prediction.

1) GENERATIVE MODEL

To learn better user and item features, two variational neural networks are built. The generative process of CVDL is similar to the deep latent Gaussian model [22]. For each user u_i, the generative model starts by sampling a $K-$dimensional latent representation $z_u \sim N(0, \mathbb{I}^K)$ from a standard Gaussian prior. The sample variable $X{\sim}p_\theta(X|z_u)$ is generated from its latent variable z_u through a decoder with the generative parameter θ. The $p_\theta(X|z_u)$ can be generated from a multivariate Bernoulli distribution (binary-value) or Gaussian distribution (real-value).

The generative process of user profile is defined as follows:

(1) For each layer $l \in [1, L]$ of the generative network,

a) For each column n of weight matrix W_l^d, draw

$$W_{l,n}^d \sim N(0, \lambda_w^{-1}\mathbb{I}_K)$$

b) Draw bias vector $b_l^d \sim N(0, \lambda_w^{-1}\mathbb{I}_K)$

c) For each row i of h_l^d, draw

$$h_{l,i}^d \sim N(\sigma(h_{l-1,i}^d W_l^d + b_l^d), \lambda_t^{-1}\mathbb{I}_K)$$

(2) For each X_i,

a) If X_i is binary, draw $X_i \sim B(\sigma(h_l^d W_l^d + b_{l+1}^d))$

b) If X_i is real-value, draw

$$X_i \sim N(h_l^d W_l^d + b_{l+1}^d, \lambda_X^{-1}\mathbb{I}_K)$$

where, λ_w, λ_t and λ_X are hyperparameters, h_l^d represents hidden layers of decoder. Similar to SDAE, λ_t is taken to infinity for computational efficiency.

The latent representation z_u can be drawn by a Gaussian prior distribution with zero mean and identity matrix: $z_u \sim N(0, \mathbb{I}_K)$. The user's latent representation u_i consists of latent user offset and the latent user profile vector, *i.e.* $u_i = \epsilon_i + z_{u_i}$.

The generative process of item content is a little different from that of user profile. CVDL employs additional VAE to generate latent content vectors, which integrates the item content information and its tag information as inputs and can effectively learn the latent vector. The generative process of Y is defined as follows:

(1) For each layer $l^{'} \in [1, L]$ of the generative network,

a) For each column n of weight matrix $W_{l'}^d$, draw

$$W_{l'}^d \sim N(0, \lambda_w^{-1}\mathbb{I}_K)$$

b) Draw additional weight matrix $W_s \sim N(0, \lambda_s^{-1}\mathbb{I}_K)$

c) Draw bias vector $b_{l'} \sim N(0, \lambda_w^{-1}\mathbb{I}_K)$

d) Draw additional bias vector $b_s \sim N(0, \lambda_s^{-1}\mathbb{I}_K)$

e) For each row j of $h_{l'}^d$, draw

$$h_{l',j}^d \sim N(\sigma(h_{l'-1,j}^d W_{l'}^d + b_{l'} + TW_s + b_s), \lambda_t^{-1}\mathbb{I}_K)$$

(2) For each Y_j,

a) If Y_j is binary, draw $Y_j \sim B(\sigma(h_{l'}^d W_{l'}^d + b_{l'+1}))$

b) If Y_j is real-value, draw

$$Y_j \sim N(h_{l'}^d W_{l'}^d + b_{l'+1}, \lambda_X^{-1}\mathbb{I}_K)$$

(3) For each T_s,

a) If T_s is binary, draw $Y_s \sim B(\sigma(h_{l'}^d W_s + b_{s+1}))$

b) If T_s is real-value, draw

$$Y_s \sim N(h_{l'}^d W_s + b_{s+1}, \lambda_s^{-1}\mathbb{I}_K)$$

where, λ_s is a hyperparameter, $h_{l'}^d$ denotes hidden layers of decoder. Then, the item latent representation v_j can be denoted by $v_i = \epsilon_j + z_{v_j}$.

2) INFERENCE MODEL

The inference model is an encoder network corresponding to the one in the generative model. For user, the inference process is to approximate the intractable posterior distribution $p_\theta(z_u|X)$ which is determined by the generative network. Using the Stochastic Gradient Variational Bayes (SGVB) estimator, the posterior of latent user profile variable z_u can be approximated by a tractable variational distribution $q_\phi(z_u|X)$.

$$q_\phi(z_u \mid X_i) = N\left(\mu_\phi(X_i), \mathrm{diag}\left(\sigma_\phi^2(X_i)\right)\right) \tag{1}$$

where, $\mu_\phi \in R^K$ and $\sigma_\phi^2 \in R^K$ are the mean value and standard deviation of the approximate posterior respectively, which are non-linear functions of X_i and the variational parameter ϕ, and they are inference outputs.

Similar to [26], the z_u inference process is defined as follows:

(1) For each layer l of the inference model,

a) For each column n of weight matrix W_l^e, draw

$$W_{l,n}^e \sim N(0, \lambda_w^{-1}\mathbb{I}_K)$$

b) Draw bias vector $b_l^e \sim N(0, \lambda_w^{-1}\mathbf{I}_K)$

c) For each row i of h_l^e, draw

$$h_{l,j}^e \sim N(\sigma(h_{l-1,j}^e W_l^e + b_l^e), \lambda_s^{-1}\mathbb{I}_K)$$

(2) For each user u_i

a) Draw latent mean vector

$$\mu_i \sim N(h_l^e W_\mu^e + b_\mu^e, \lambda_s^{-1}\mathbb{I}_K)$$

b) Draw latent covariance vector

$$\log \sigma_i^2 \sim N(h_l^e W_\sigma^e + b_\sigma^e, \lambda_s^{-1}\mathbb{I}_K)$$

c) Draw latent content vector $z_u \sim N(\mu_i, \mathrm{diag}(\sigma_i^2))$

As explained in [26], the evidence lower bound (ELBO) for X_i can be estimated using SGVB estimator:

$$\begin{aligned}
&\mathcal{L}(\theta, \phi; X_i)\\
&\quad = \mathrm{E}_{q_\phi(z_u|X_i)}\left[\log p(u_i|z_u) + \log p_\theta(X_i|z_u)\right]\\
&\quad\quad - \beta \cdot KL\left(q_\phi(z_u|X_i) \| p(z_u)\right)\\
&\quad \approx \log p\left(u_i \middle| z_{u,l}\right) + \frac{1}{L}\sum_{l=1}^{L} \log p_\theta\left(X_i \middle| z_{u,l}\right)\\
&\quad\quad - \beta \cdot KL\left(q_\phi(z_u|X_i) \| p(z_u)\right)
\end{aligned} \tag{2}$$

$$\begin{aligned}
&KL\left(q_\phi(z_u|X_i) \| p(z_u)\right)\\
&\quad = \frac{1}{2}\sum_{i=1}^{M}\left(\mu_i^2 + \sigma_i^2 - \log \sigma_i^2 - 1\right)
\end{aligned} \tag{3}$$

$$z_{u_i,l} = \mu_i + \sigma_i \otimes \varepsilon_{i,l} \tag{4}$$

where, KL is the Kullback-Leibler divergence, $\beta \in [0, 1]$ is a parameter to control the regularization strength for addressing the posterior collapse problem [30], $\varepsilon_{i,l} \sim N(0, 1)$, and $\otimes$ is the element-wise product.

The inference process of item content is similar to user profile inference process, and the ELBO for item network can be derived in the same way:

$$\begin{aligned}
&\mathcal{L}(\theta, \phi; Y_j, T_s)\\
&\quad= \mathrm{E}_{q_\phi(z_v|Y_j,T_s)}\left[\log p\left(v_j\left|z_v\right.\right) + \log p_\theta\left(Y_j, T_s\left|z_v\right.\right)\right]\\
&\quad\quad - \beta\cdot KL\left(q_\phi\left(z_v\left|Y_j, T_s\right.\right)\|p\left(z_v\right)\right)\\
&\quad\approx \log p\left(v_j\left|z_{v_j,l'}\right.\right) + \frac{1}{L}\sum_{l=1}^{L}\log p_\theta(Y_j, T_s\left|z_{v_j,l'}\right.)\\
&\quad\quad - \beta\cdot KL\left(q_\phi\left(z_v\left|Y_j, T_s\right.\right)\|p\left(z_v\right)\right) \qquad (5)
\end{aligned}$$

Then, the generative process of the CVDL model is defined as follows:

(1) For each user u_i,
  a) Draw latent user offset $\epsilon_i \sim N(0, \lambda_u^{-1}\mathbb{I}_K)$)
  b) Set latent user vector as $u_i = \epsilon_i + z_{u_i}$
(2) For each item v_j,
  a) Draw latent item offset $\epsilon_j \sim N(0, \lambda_v^{-1}\mathbb{I}_K)$)
  b) Set latent item vector as $v_j = \epsilon_j + z_{v_j}$
(3) For each user-item pair (u_i, v_j), draw rating

$$R_{ij} \sim N(u_i^\top v_j, c_{ij}^{-1})$$

C. OPTIMIZATION

Through the CVDL model, we utilize maximum a poster probability estimator to learn parameters of our model. The objective function includes three parts: the latent loss, the regularization loss and the KL loss, shown as follows.

$$\begin{aligned}
\mathcal{L} = &-\sum_{i,j}\frac{C_{ij}}{2}\left(R_{ij} - u_i^\top v_j\right)^2\frac{\lambda_w}{2}\sum_{l=1}^{L}\left(\|W_l\|_\mathrm{F}^2 + \|b_l\|_\mathrm{F}^2\right)\\
&-\frac{\lambda_s}{2}\sum_{l=1}^{L}\left(\|W_s\|_\mathrm{F}^2 + \|b_s\|_\mathrm{F}^2\right) - \frac{\lambda_u}{2}\sum_{i}^{M}\mathrm{E}_{q_\phi(z_u|X_i)}\|u_i - z_i\|_\mathrm{F}^2\\
&+\mathrm{E}_{q_\phi(z_u|X_i)}\left[\log p\left(X_i\left|z_u\right.\right)\right] - \beta\cdot KL\left(q_\phi\left(z_u\left|X_i\right.\right)\|p\left(z_u\right)\right)\\
&-\frac{\lambda_v}{2}\sum_{j}^{N}\mathrm{E}_{q_\phi(z_v|Y_j,T_s)}\left\|v_j - z_j\right\|_\mathrm{F}^2\\
&+\mathrm{E}_{q_\phi(z_v|Y_j,T_s)}\left[\log p\left(Y_j, T_s\left|z_v\right.\right)\right]\\
&-\beta\cdot KL\left(q_\phi\left(z_v\left|Y_j, T_s\right.\right)\|p\left(z_v\right)\right) \qquad (6)
\end{aligned}$$

where, C_{ij} is the confidence parameter for R, and $||\cdot||_\mathrm{F}$ denotes the Frobenius norm.

This objective function can be optimized by taking the gradient $\mathcal{L}$ with respect to u_i and v_j and setting them to zero. The update equations are derived as follows:

$$\frac{\partial\mathcal{L}}{\partial u_i} = \sum_{i\in\mathcal{R}_j^+} C_{ij}\left(R_j - u_i^\top v_j\right)v_j - \lambda_u\mathrm{E}_{q_\phi(z_u)}\left[u_i - z_{u_i}\right] \qquad (7)$$

$$\frac{\partial\mathcal{L}}{\partial v_j} = \sum_{i\in\mathcal{R}_j^-} C_{ij}\left(R_j - u_i^\top v_j\right)u_i - \lambda_v\mathrm{E}_{q_\phi(z_v)}\left[v_j - z_{v_j}\right] \qquad (8)$$

where R_i^+ denotes the items that u_i has rated, and R_j^- represents the users who have rated v_j.

Let the gradients with respect to u_i and v_j be zero, we have

$$u_i \leftarrow \left(VC_iV^\top + \lambda_u\mathbb{I}_K\right)^{-1}\left(VC_iR_i + \lambda_u\mathrm{E}_{q_\phi(z_u)}\left[z_{u_i}\right]\right) \qquad (9)$$

$$v_j \leftarrow \left(UC_jU^\top + \lambda_v\mathbb{I}_K\right)^{-1}\left(UC_jR_j + \lambda_v\mathrm{E}_{q_\phi(z_v)}\left[z_{v_j}\right]\right) \qquad (10)$$

where C_i is the diagonal matrix with $C_{ij}(j = 1, 2, .., N)$ as its diagonal elements and $R_i = (R_{ij})_{j=1}^N$ for user u_i.

For each item v_j, C_j and R_j are similarly defined. It can be easily found that the $\mathrm{E}_{q_\phi(z)}[z]$ equals to μ produced by the encoder. Given U and V, the gradient with respect to μ and σ of z can be computed as follows:

$$\begin{aligned}
&\frac{\partial\mathcal{L}(\theta, \phi; X_i)}{\partial\mu_i}\\
&\quad\approx \frac{\lambda_u}{L}\sum_{l=1}^{L}\left(u_i - z_{u_i,l}\right) + \frac{\log p_\theta\left(X_i\left|z_{u_i,l}\right.\right)}{\partial z_{u_i,l}} - \beta\cdot\mu_i \qquad (11)\\
&\frac{\partial\mathcal{L}(\theta, \phi; Y_j, T_s)}{\partial\mu_j}\\
&\quad\approx \frac{\lambda_v}{L}\sum_{l=1}^{L}\left(v_j - z_{v_j,l'}\right) + \frac{\log p_\theta\left(Y_j, T_s\left|z_{v_j,l'}\right.\right)}{\partial z_{v_j,l'}} - \beta\cdot\mu_j\\
&\qquad\qquad (12)\\
&\frac{\partial\mathcal{L}(\theta, \phi; X_i)}{\partial\sigma_i}\\
&\quad\approx \left[\frac{\lambda_u}{L}\sum_{l=1}^{L}\left(u_i - z_{u_i,l}\right) + \frac{\log p_\theta\left(X_i\left|z_{u_i,l}\right.\right)}{\partial z_{u_i,l}}\right]\otimes\varepsilon_l\\
&\quad\quad -\beta\cdot\left(\sigma_i - \sigma_i^{-1}\right) \qquad (13)\\
&\frac{\partial\mathcal{L}(\theta, \phi; Y_j, T_s)}{\partial\sigma_j}\\
&\quad\approx \left[\frac{\lambda_v}{L}\sum_{l=1}^{L}\left(v_j - z_{v_j,l'}\right) + \frac{\log p_\theta\left(Y_j, T_s\left|z_{v_j,l'}\right.\right)}{\partial z_{v_j,l'}}\right]\otimes\varepsilon_{l'}\\
&\quad\quad - \beta\cdot\left(\sigma_j - \sigma_j^{-1}\right) \qquad (14)
\end{aligned}$$

where $\log p_\theta\left(X_i\left|z_{u_i,l}\right.\right)$, $\log p_\theta\left(Y_j, T_s\left|z_{v_j,l'}\right.\right)$ are determined by X_i and (Y_j, T_s), respectively. If X_i and (Y_j, T_s) are in binary,$p_\theta(X|z_u)$ and $p_\theta(Y, T|z_v)$ are Bernoulli distributions; if X_i and (Y_j, T_s) are in categorical, $p_\theta(X|z_u)$ and $p_\theta(Y, T|z_v)$ follow Categorical distributions; if X_i and (Y_j, T_s) are in real-valued, $p_\theta(X|z_u)$ and $p_\theta(Y, T|z_v)$ are Gaussian. Consequently, the weight gradients of decoder and encoder can be obtained by using back propagation.

D. PREDICTION

After the optimal parameters are learned, CVDL can be employed for in-matrix (non cold-start) and out-matrix (cold-start) prediction. Assumed that O is the observed rating data, and both types of predictions can be evaluated by point estimation. For in-matrix prediction, we have

$$\begin{aligned}
\mathrm{E}\left[R_{ij}\left|\mathrm{O}\right.\right] &\approx \mathrm{E}\left[u_i\left|\mathrm{O}\right.\right]^\top\mathrm{E}\left[u_j\left|\mathrm{O}\right.\right]\\
&= \left(\mathrm{E}\left[\epsilon_i\left|\mathrm{O}\right.\right] + \mathrm{E}\left[z_{u_i}\left|\mathrm{O}\right.\right]\right)^\top\\
&\quad\times\left(\mathrm{E}\left[\epsilon_j\left|\mathrm{O}\right.\right] + \mathrm{E}\left[z_{v_j}\left|\mathrm{O}\right.\right]\right) \qquad (15)
\end{aligned}$$

$$\begin{aligned} R_{ij} &\approx u_i^\top v_j = (\epsilon_i + \mathrm{E}\left[z_{u_i}|\mathrm{O}\right])(\epsilon_j + \mathrm{E}\left[z_{v_j}|\mathrm{O}\right]) \\ &= (\epsilon_i + \mu_i)^\top (\epsilon_j + \mu_j) \end{aligned} \tag{16}$$

For out-matrix prediction, the item is new and has not been rated by other users or the user is new and has not rated any item, which means $\mathrm{E}\left[\epsilon_j\right] = 0$ or $\mathrm{E}\left[\epsilon_i\right] = 0$, so the rating predictions can be calculated by

$$\begin{aligned} \mathrm{E}\left[R_{ij}|\mathrm{O}\right]^{cold-item} &\approx \mathrm{E}\left[u_i|\mathrm{O}\right]^\top \mathrm{E}\left[v_j|\mathrm{O}\right] \\ &= \left(\mathrm{E}\left[\epsilon_i|\mathrm{O}\right] + \mathrm{E}\left[z_{u_i}|\mathrm{O}\right]\right)^\top \mathrm{E}\left[z_{v_j}|\mathrm{O}\right] \end{aligned} \tag{17}$$

$$\begin{aligned} \mathrm{E}\left[R_{ij}|\mathrm{O}\right]^{cold-user} &\approx \mathrm{E}\left[u_i|\mathrm{O}\right]^\top \mathrm{E}\left[v_j|\mathrm{O}\right] \\ &= \mathrm{E}\left[z_{u_i}|\mathrm{O}\right]^\top \left(\mathrm{E}\left[\epsilon_j|\mathrm{O}\right] + \mathrm{E}\left[z_{v_j}|\mathrm{O}\right]\right) \\ R_{ij}^{cold-item} &\approx (\epsilon_i + \mu_i)^\top \mu_j \\ R_{ij}^{cold-user} &\approx \mu_i^\top (\epsilon_j + \mu_j) \end{aligned} \tag{18}$$

IV. EXPERIMENTAL RESULTS

A. EXPERIMENTAL SETTINGS

1) DATASETS

In this section, three real-world datasets are selected to evaluate our model. First two are public datasets from CiteULike which are *citeulike-a* and *citeulike-t*. The third one is an anonymized healthcare dataset collected from 16 hospital in Georgia State (denoted by GHC), which includes over 2 million transactions and over 50 distinct attributes over 7 years. Table 2 summarizes the characteristics of CiteULike and GHC datasets.

TABLE 2. Statistics of citeULike and GHC datasets.

	citeulike-a	*citeulike-t*	GHC
Num. of user	5,551	7,947	452,116
Num. of item	16,980	25,975	6,872
Num. of rating	204,986	134,860	2,998,351
Num. of tag	7836	8311	18,692
Num. of vocabulary	8,000	20,000	---
Sparsity	99.78%	99.93%	99.90%
User Features	---	---	Demographics, Tag
Item Features	Title, Abstract	Title, Abstract	Demographics, Education, Specialties

The two CiteULike*datasets* are from the work [11]. The *citeulike-a* contains 5,551 users and 16,980 articles with 204,986 observed user-item ratings. Users with fewer than 10 ratings are not included and the sparsity of *citeulike-a* is 99.78%. *citeulike-t* includes 7,947 users and 25,975 items with 134,860 observed ratings, and its sparsity is 99.93%, which is much sparser than the former one. Similar to *citeulike-a*, users with fewer than 3 ratings are excluded. Each item in the two datasets has a title and abstract, and the content information is the concatenation of the titles and abstracts. The vocabulary for each dataset is selected according to the tf-idf value of each word. The *citeulike-a* has a vocabulary size of 8,000, while the *citeulike-t* has a vocabulary size of 20,000. Each article is represented with a bag-of-words vector and all the content vectors are then normalized over the maximum occurrences of each word in all articles.

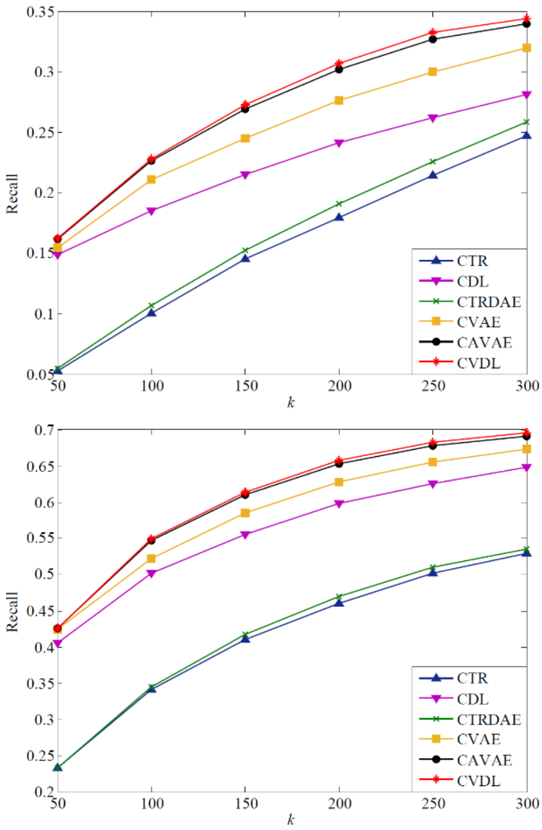

FIGURE 3. Performance comparison of all methods on *citeulike-a* in the sparse and dense settings.

The anonymized dataset GHC contains the transactions of healthcare services for treat a clinical condition or procedure. Patients are assigned a unique ID across the healthcare network located in Georgia State. We first preprocess the data identify consistent IDs of patient and doctor in healthcare network, and then derive one interaction between a patient and a doctor. After data cleansing, we have 35 million interactions between around 1.1 million patients and 7,960 doctors. For each patient, the basic demographic characteristics such as gender, age, residence zipcode, *etc.* and tags of clinical condition are obtained. For each doctor, the demographic characteristics, education, and their medical specialties are collected. The interactions between a patient and a doctor are regarded as reviews associated with a rating ranging from 1 to 5, and patients with less than 5 reviews and doctors that have been rated by less than 10 users are deleted. Then, the rating matrix is binarized using value 3 as a threshold. The final dataset obtains 452,116 patients (users), 6,872 doctors (items) and 2,998,351 ratings, of which the sparsity is 99.90%.

2) BASELINES AND EVALUATION METRICS

To evaluate our CVDL, five representative CTR models are selected as Baselines.

(1) CTR [8] is a probabilistic topic modeling based collaborative filtering recommendation model that incorporates both ratings and item contents.

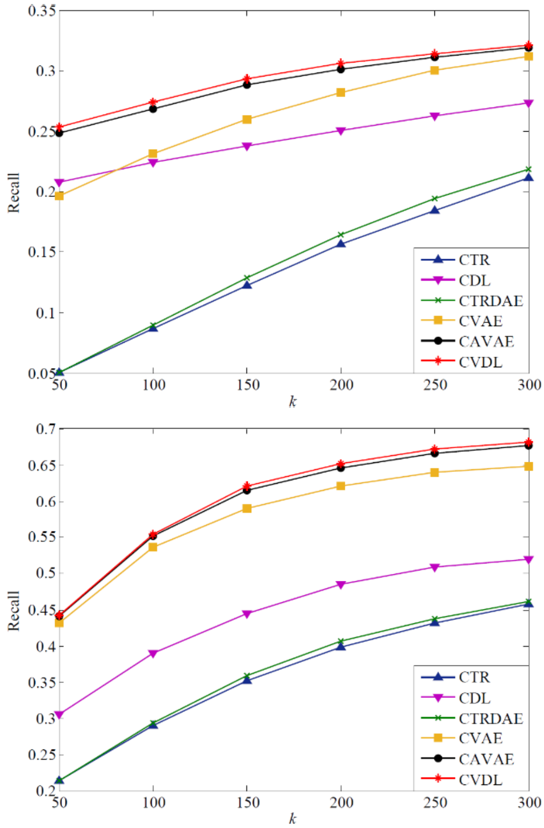

FIGURE 4. Performance comparison of all methods on *citeulike-t* in the sparse and dense settings.

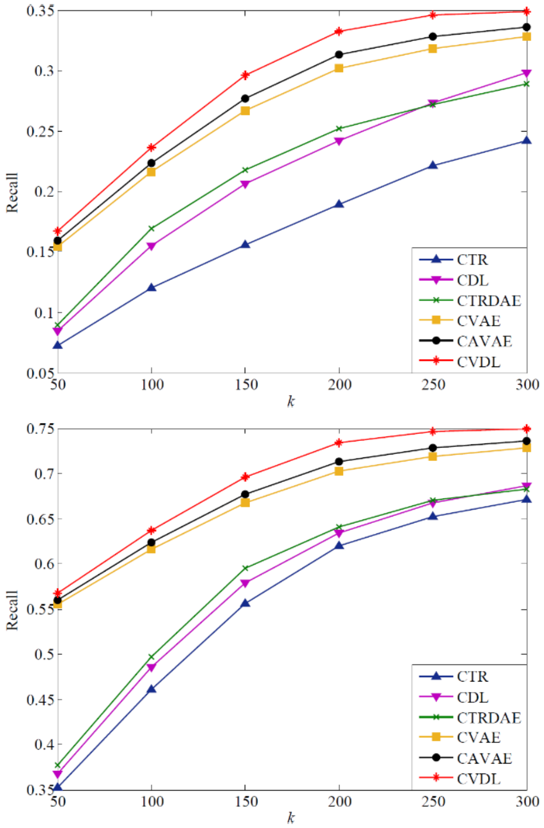

FIGURE 5. Performance comparison of all methods on GHC in the sparse and dense settings.

(2) CDL [11] is a probabilistic joint learning model of CTR and SDAE, which can achieve promising performance.

(3) CTRDAE [28] is similar model to CDL, a combination of CTR and DAE for exploring the effects of communities on learning users' preferences.

(4) CVAE [23] is a Bayesian generative model that jointly models CTR and VAE to bridge auxiliary information together with deep architecture.

(5) CAVAE [24] is the improved version of CVAE, which integrates CTR with additional VAE to extract effective latent vector from side information.

To evaluate the performance of our models, Recall [11] is adopted as the evaluation metric. The predictive ratings of candidate items are sorted, and the first k items are recommend to the target user. The Recall is defined in following equation.

$$Recall@k = \frac{\text{The items the user likes among the top } k}{\text{The items the user likes}} \tag{19}$$

3) PARAMETER SETTINGS

To comparatively evaluate our model, we set the same dimensionality of latent space $K = 50$ and the same tuning confidence parameters $a = 1$ and $b = 0.01$ for all algorithms. Then we use fivefold cross validation to find the optimal hyperparameters for CTR, CDL, CTRDAE, CVAE, CAVAE and CVDL, and the optimal parameters for each method are listed in Table 3.

TABLE 3. Parameter settings of all methods.

Methods	Optimal Parameter Settings
CTR	λ_u=0.1, λ_v=1
CDL	λ_u=0.01, λ_v=10, λ_n=1000, λ_w=0.0001
CTRDAE	λ_u=0.1, λ_v=100, λ_n=0.1
CVAE	λ_u=0.1, λ_v=10, λ_w=0.0001, λ_r=10
CAVAE	λ_u=0.1, λ_v=10, λ_w=0.0001
CVDL	λ_u=0.1, λ_v=10, λ_w=0.0001, β=0.2

For CTR, it is found that it can achieve best performance when $\lambda_u = 0.1$ and $\lambda_v = 1$, and it is first pretrained with LDA to get the initial topic proportions and CTR is performed to jointly learn U, V and topic proportions iteratively. For CDL, the best performance is achieved with $\lambda_u = 0.01$, $\lambda_v = 10$, $\lambda_n = 1000$ and $\lambda_w = 0.0001$. For CTRDAE, the autoencoder is pretrained with a single layer before joint-learning with CTR, and the model receives the best performance as $\lambda_u = 0.1$, $\lambda_v = 100$ and $\lambda_n = 0.1$. The ratio λ_n:λ_u is found to be 1:1 for the best performance. For CVAE, CAVAE and our CVDL, the models gain the best predictive accuracy when $\lambda_u = 0.1$, $\lambda_v = 10$ and $\lambda_w = 0.0001$, and the specified parameter λ_r is set to 10 for CVAE. To select the parameter β,

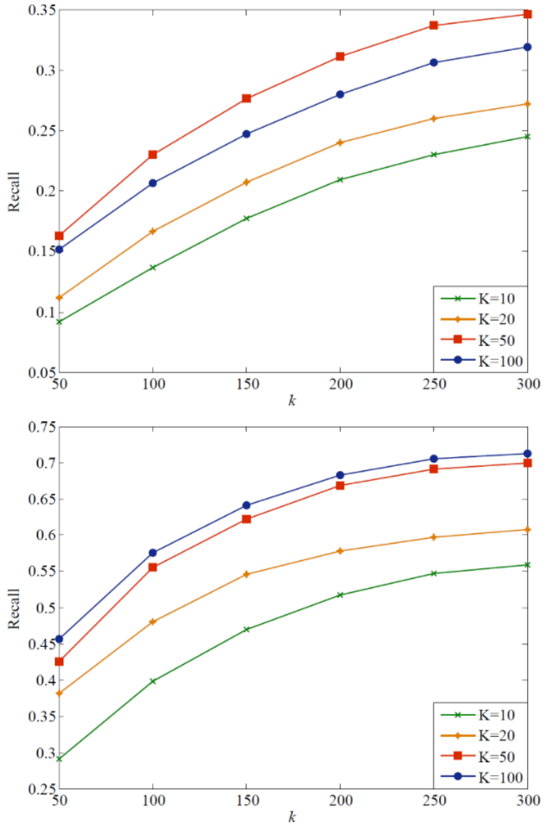

FIGURE 6. Performance comparison of different K on *citeulike-a* in sparse and dense settings.

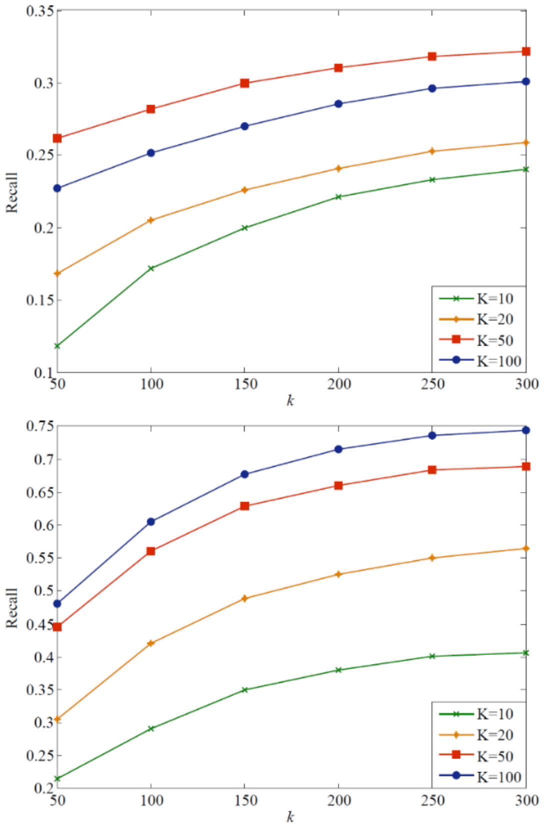

FIGURE 7. Performance comparison of different K on *citeulike-t* in sparse and dense settings.

the training procedure starts with 0 and gradually increases to 1, and β is set to 0.2 for best performance.

B. EXPERIMENTAL RESULTS

1) OVERALL PERFORMANCE COMPARISON

In our experiments, each dataset is split into two parts: training datasets and testing datasets. For the training set, experiments are carried out with a setting of 80% random sample of each user ratings, and the rest of user ratings (20%) are used for testing. We randomly select P items associated with each user to form the training set and use all the rest of the dataset as the test set. To evaluate and compare the models under both sparse and dense settings, we set the parameter P to 1 and 10, respectively. For each P, we conduct the five independent evaluations with different randomly selected training sets and get the average performance.

Figures 3-5 show the *Recall@k* results that compare CDL, CTR, CTRDAE, CVAE, CAVAE and CVDL on three datasets under the sparse setting ($P = 1$) and the dense setting ($P = 10$).

From Figures 3 and 4, it is obvious that CVDL outperforms all the other baselines on both CiteULike datasets with different settings. The comparison results from two CiteULike datasets indicate that CVDL, CAVAE, CVAE and CDL achieve better performance than CTR and CTRDAE, which is because that CTR's latent representation capability is limited due to the LDA, and CTR cannot learn potential representation effectively, especially when side information is very sparse. In addition, CVDL, CAVAE and CVAE show consistent performance on two CiteULike datasets, while CDL only has better performance on *citeulike-t* than *citeulike-a*. These can be explained that CVDL, CAVAE and CVAE employ VAE for seeking probabilistic latent content variable model, which can learn latent vectors effectively. By contrast, CDL utilizes SDAE to learn latent content vectors by corrupting input data, easily leading to data overfitting.

From Figure 5, it can be easily found that the proposed CVDL achieves the best performance on GHC dataset, which sustains the powerful latent representation learning capability of our model. We also found that CTRDAE outperforms CTR and CDL when k is no larger than 250, and it can be explained that CTRDAE learn user profile representations jointly with item contents for better regularizing latent user and item factors, leading to better performance. When the training data comes denser, the performance CTRDAE becomes worse than CDL, due to the higher diversity in users preferences. To focus on the performance comparison of the VAE-based models, it is clear that CVDL achieves better performance than CVAE and CAVAE. This can be explained that CVDL integrates both user and item side information

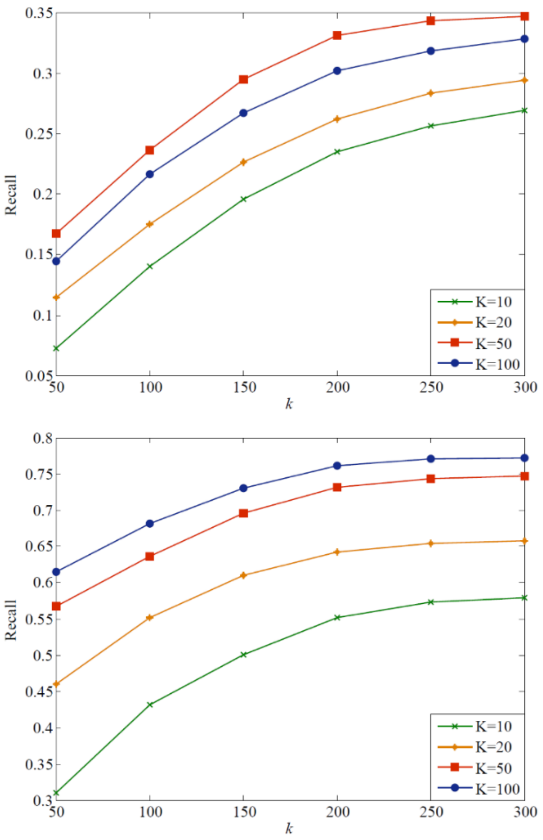

FIGURE 8. Performance comparison of different K on GHC in sparse and dense settings.

as inputs to learn latent vector compared with CVAE and CAVAE, which can extract more robust latent vectors from content/profile. Therefore, the Recall metric proves the effectiveness of our CVDL.

2) SENSITIVITY ANALYSIS

To evaluate the sensitivity to hyperparameters, the similar procedure [18] is conducted to evaluate the performance of our CVDL with different K values based on recall for all datasets with sparse and dense settings. The K indicates the dimensionality of latent space, and different K values make a difference between the latent vector of content/profile and the latent variable for PMF.

Figures 6-8 shows the results of CVDL on three datasets for different K in both sparse setting and dense setting. The influence of K depends on three parts, the latent variables for the representation of item content and user profile, and the latent variable for the matrix factorization model. It is clear that the larger K enables CVDL to learn a better representation from item content and user profile, which leads to upgraded prediction performance. However, if K goes large enough, its influence becomes trivial since the representation capability is enough for modeling item content and user profile. Comparing the influence of K in the sparse and dense settings, it is evident that the larger K has greater impact in the dense setting, which is mainly because the denser ratings can facilitate the process of variational inference.

V. CONCLUSION

In this paper, we propose a hybrid collaborative deep learning model (CVDL) for healthcare recommendation, which jointly models the generation of item content and user profile while extracting the implicit relationships between items and users collaboratively. On the one hand, the proposed CVDL can be considered as a Bayesian probabilistic generative model, and its variational inference is deduced from a stochastic gradient variational Bayesian model. CVDL unifies the collaborative information, item content and user profile through deep learning model and graphical model, which leads to robust recommending performance. On the other hand, our CVDL can unify multimedia in different forms for recommendation, due to its inference of stochastic distribution in latent space instead of observation space. Experimental results have shown the proposed CVDL can significantly outperform the current CTR approaches for recommendation jointly with item content and user profile, with more robust performance, especially on the healthcare dataset.

CVDL is proposed by utilizing MLP as the inference and generation models, which can also fit into other deep learning models, depending on the data type of additional information. In future, we try to investigate different mappings between healthcare communities and relevant topics, and plan to incorporate knowledge graph to obtain more side information to further improve the precision of HRS.

REFERENCES

[1] T. Bodenheimer and H. H. Pham, "Primary care: Current problems and proposed solutions," *Health Affairs*, vol. 29, no. 5, pp. 799–805, 2010.

[2] K. Gottlieb, I. Sylvester, and D. Eby, "Transforming your practice: What matters most," *Family Pract. Manage.*, vol. 15, no. 1, pp. 32–38, 2008.

[3] M. Wiesner and D. Pfeifer, "Health recommender systems: Concepts, requirements, technical basics and challenges," *Int. J. Environ. Res. Public Health*, vol. 11, no. 3, pp. 2580–2607, 2014.

[4] Y. Shi, M. Larson, and A. Hanjalic, "Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges," *ACM Comput. Surv.*, vol. 47, pp. 3:1–3:45, May 2014.

[5] A. Mnih and R. R. Salakhutdinov, "Probabilistic matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2008, pp. 1257–1264.

[6] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *IEEE Comput.*, vol. 42, no. 8, pp. 30–37, Aug. 2009.

[7] J. Zhong and X. Li, "Unified collaborative filtering model based on combination of latent features," *Expert Syst. Appl.*, vol. 37, no. 8, pp. 5666–5672, 2010.

[8] C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2011, pp. 448–456.

[9] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," in *Proc. 25th Conf. Uncertainty Artif. Intell.*, 2009, pp. 452–461.

[10] W. Pan, H. Zhong, C. Xu, and Z. Ming, "Adaptive Bayesian personalized ranking for heterogeneous implicit feedbacks," *Knowl.-Based Syst.*, vol. 73, pp. 173–180, Jan. 2015.

[11] H. Wang, N. Wang, and D. Y. Yeung, "Collaborative deep learning for recommender systems," in *Proc. 21st ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2015, pp. 1235–1244.

[12] S. Li, J. Kawale, and Y. Fu, "Deep collaborative filtering via marginalized denoising auto-encoder," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, 2015, pp. 811–820.

[13] H. Ying, L. Chen, Y. Xiong, and J. Wu, "Collaborative deep ranking: A hybrid pair-wise recommendation algorithm with implicit feedback," in *Proc. 20th Pacific–Asia Conf. Knowl. Discovery Data Mining*, 2016, pp. 555–567.

[14] X. Dong, L. Yu, Z. Wu, Y. Sun, L. Yuan, and F. Zhang, "A hybrid collaborative filtering model with deep structure for recommender systems," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 1309–1315.

[15] S. Purushotham, Y. Liu, and C. C. J. Kuo, "Collaborative topic regression with social matrix factorization for recommendation systems," in *Proc. 29th Int. Conf. Mach. Learn.*, 2012, pp. 9–15.

[16] C. Chen, X. Zheng, Y. Wang, F. Hong, and Z. Lin, "Context-aware collaborative topic regression with social matrix factorization for recommender systems," in *Proc. 28th AAAI Conf. Artif. Intell.*, 2014, pp. 9–15.

[17] J. H. Kang and K. Lerman, "LA-CTR: A limited attention collaborative topic regression for social media," in *Proc. 27th AAAI Conf. Artif. Intell.*, 2013, pp. 1128–1134.

[18] H. Wang and W. J. Li, "Relational collaborative topic regression for recommender systems," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 5, pp. 1343–1355, May 2015.

[19] H. Wu, K. Yue, Y. Pei, B. Li, Y. Zhao, and F. Dong, "Collaborative topic regression with social trust ensemble for recommendation in social media systems," *Knowl.-Based Syst.*, vol. 97, pp. 111–122, Apr. 2016.

[20] A. A. Sánchez-Escalona and E. Góngora-Leyva, "Artificial neural network modeling of hydrogen sulphide gas coolers ensuring extrapolation capability," *Math. Model. Eng. Problems*, vol. 5, no. 4, pp. 348–356, 2018.

[21] M. K. Benkaddour and A. Bounoua, "Feature extraction and classification using deep convolutional neural networks, PCA and SVC for face recognition," *Traitement Signal*, vol. 34, nos. 1–2, pp. 77–91, 2017.

[22] D. P. Kingma and M. Welling. (2013). "Auto-encoding variational Bayes." [Online]. Available: https://arxiv.org/abs/1312.6114

[23] X. Li and J. She, "Collaborative variational autoencoder for recommender systems," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2017, pp. 305–314.

[24] M. He, Q. Meng, and S. Zhang, "Collaborative additional variational autoencoder for top-N recommender systems," *IEEE Access*, vol. 7, pp. 5707–5713, 2019.

[25] W. Lee, K. Song, and I. C. Moon, "Augmented variational autoencoders for collaborative filtering with auxiliary information," in *Proc. ACM Conf. Inf. Knowl. Manage.*, 2017, pp. 1139–1148.

[26] D. Liang, R. G. Krishnan, M. D. Hoffman, and T. Jebara, "Variational autoencoders for collaborative filtering," in *Proc. World Wide Web Conf.*, 2018, pp. 689–698.

[27] M. D. Hoffman and M. J. Johnson, "Elbo surgery: Yet another way to carve up the variational evidence lower bound," in *Proc. Workshop Adv. Approx. Bayesian Inference*, 2016, pp. 1–4.

[28] T. T. Nguyen and H. W. Lauw, "Collaborative topic regression with denoising autoencoder for content and community co-representation," in *Proc. ACM Conf. Inf. Knowl. Manage.*, 2017, pp. 2231–2234.

[29] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Comput. Surv.*, vol. 52, no. 1, p. 5, 2019.

[30] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio, "Generating sentences from a continuous space," in *Proc. 20th SIGNLL Conf. Comput. Natural Lang. Learn.*, 2016, pp. 10–21.

XIAOYI DENG received the Ph.D. degree in management science and engineering from the Dalian University of Technology, in 2012. He is currently an Associate Professor with Business School, Huaqiao University, China. His current research interests are recommender systems, data mining, and knowledge discovery. Since 2010, he has been a member of the China Computer Federation (CCF) and the Systems Engineering Society of China (SESC).

FEIFEI HUANGFU received the M.A. degree in linguistics from Lancaster University, in 2009. She is currently an Assistant Professor with the College of Foreign Languages, Huaqiao University. She is also a member of the China Computer Federation.