

## 用于协同过滤的序列解耦变分自编码器

伍美霖, 黄佳进, 秦进

### 引用本文

伍美霖, 黄佳进, 秦进. 用于协同过滤的序列解耦变分自编码器[J]. 计算机科学, 2022, 49(12): 163-169.

WU Mei-lin, HUANG Jia-jin, QIN Jin. [Disentangled Sequential Variational Autoencoder for Collaborative Filtering](#) [J]. Computer Science, 2022, 49(12): 163-169.

---

### 相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

#### Similar articles recommended (Please use Firefox or IE to view the article)

#### [基于TPH-YOLOv5和小样本学习的害虫识别方法](#)

Pest Identification Method Based on TPH-YOLOv5 Algorithm and Small Sample Learning  
计算机科学, 2022, 49(12): 257-263. <https://doi.org/10.11896/jsjcx.221000203>

#### [基于改进Sigmoid卷积神经网络的手写体数字识别](#)

Handwritten Numeral Recognition Based on Improved Sigmoid Convolutional Neural Network  
计算机科学, 2022, 49(12): 244-249. <https://doi.org/10.11896/jsjcx.211000179>

#### [基于反事实思考的视觉问答方法](#)

Visual Question Answering Method Based on Counterfactual Thinking  
计算机科学, 2022, 49(12): 229-235. <https://doi.org/10.11896/jsjcx.220600038>

#### [深度学习方法在二维人体姿态估计的研究进展](#)

Research Progress of Deep Learning Methods in Two-dimensional Human Pose Estimation  
计算机科学, 2022, 49(12): 219-228. <https://doi.org/10.11896/jsjcx.210900041>

#### [面向深度卷积神经网络的小目标检测算法综述](#)

Small Object Detection Based on Deep Convolutional Neural Networks:A Review  
计算机科学, 2022, 49(12): 205-218. <https://doi.org/10.11896/jsjcx.220500260>

# 用于协同过滤的序列解耦变分自编码器

伍美霖<sup>1</sup> 黄佳进<sup>2</sup> 秦进<sup>1</sup>

<sup>1</sup> 贵州大学计算机科学与技术学院 贵阳 550025

<sup>2</sup> 北京工业大学国际 WIC 研究院 北京 100000

(gs.mlwu19@gzu.edu.cn)

**摘要** 推荐模型通常使用用户的历史行为来获得用户偏好表示,以产生推荐。大多数方法学习到的用户表示会把不同的偏好因素纠缠在一起,而解耦学习的方法可以用于分解用户的行为特征。为此,文中提出了一个基于变分自编码器的框架 DSVAECF,用于从用户历史行为中分解静态和动态偏好因素。首先,DSVAECF 模型的两个编码器分别使用多层感知机和循环神经网络对用户行为进行历史行为建模,以此得到用户的静态和动态偏好表示;然后,将拼接的静态和动态偏好表示视为用户偏好的解耦表示,并将其输入解码器来捕获用户的决策,并重构出用户行为。在模型训练阶段,一方面最大化重构的用户行为与真实用户行为之间的互信息来学习模型参数;另一方面通过最小化解耦表示与其先验分布间的差异来保留模型的生成能力。在 Amazon 和 MovieLens 两个数据集上的实验结果表明,与基准方法相比,DSVAECF 在归一化折损累计增益、精确率和召回率上都有显著的提升,拥有更好的推荐性能。

**关键词**: 变分自编码器;深度学习;序列建模;解耦学习;协同过滤

中图法分类号 TP391

## Disentangled Sequential Variational Autoencoder for Collaborative Filtering

WU Mei-lin<sup>1</sup>, HUANG Jia-jin<sup>2</sup> and QIN Jin<sup>1</sup>

<sup>1</sup> School of Computer Science and Technology, Guizhou University, Guiyang 550025, China

<sup>2</sup> International WIC Institute, Beijing University of Technology, Beijing 100000, China

**Abstract** Recommendation models typically use user's historical behaviors to obtain user preference representations for recommendations. Most of the methods of learning user representations always entangle different preference factors, while the disentangled learning method can be used to decompose user behavior characteristics. In this paper, a variational autoencoder based framework DSVAECF is proposed to disentangle the static and dynamic factors from user's historical behaviors. Firstly, two encoders of the model use multi-layer perceptron and recurrent neural network to model the user behavior history respectively, so as to obtain the static and dynamic preference representation of the user. Then, the concatenate static and dynamic preference representations are treated as disentangled representation input decoders to capture user's decisions and reconstruct user's behavior. On the one hand, in the model training phase, DSVAECF learns model parameters by maximizes the mutual information between reconstructed user's behaviors and actual user's behaviors. On the other hand, DSVAECF minimizes the difference between disentangled representations and their prior distribution to retain the generation ability of the model. Experimental results on Amazon and MovieLens data sets show that, compared with the baselines, DSVAECF significantly improves the normalized discounted cumulative gain, recall, and precision, and has better recommendation performance.

**Keywords** Variational autoencoder, Deep learning, Sequence modeling, Disentangled learning, Collaborative filtering

## 1 引言

在各种推荐方法中,协同过滤(Collaborative Filtering, CF)<sup>[1]</sup>方法旨在使用用户的交互历史来生成推荐。其中,隐因子模型(如矩阵分解)<sup>[2]</sup>将用户和物品放入一个隐式空间中,学习用户和物品在此空间的隐向量,并用它来计算推荐

分数。近年来,隐因子模型已被广泛应用并扩展到神经网络<sup>[3]</sup>,取得了优异的性能。而变分自编码器(Variational Auto-encoder, VAE)<sup>[4-5]</sup>框架则将深度学习与概率隐因子建模相结合,同样取得了不错的效果。从本质上讲,基于变分自编码器的协同过滤将概率分布和非线性映射相结合,对用户购买历史的概率分布进行建模,并使用隐向量的先验

到稿日期:2021-12-06 返修日期:2022-03-25

基金项目:贵州省科学技术基金,黔科合基础([2020]1Y275)

This work was supported by the Guizhou Province Science and Technology Foundation([2020]1Y275).

通信作者:黄佳进(jhuang@bjut.edu.cn)

分布来对其进行正则化。

基于变分自编码的协同过滤方法可分为两类。一种假设用户历史行为的顺序对用户下一时刻的选择的影响可以被忽略,这个假设适用于建模用户长期的静态偏好和行为。另一种假设用户的选择具有时间依赖性,即用户下一时刻的选择与用户历史行为的顺序相关,这种假设适合建模用户的短期的动态偏好和行为。然而,序列数据<sup>[6]</sup>通常同时包括长期静态偏好因素和短期动态偏好因素,用户的行为会在这两种偏好的复交互作用下产生,为此需要学习用户的解耦表示(Disentangled Representations, DR)<sup>[7]</sup>。在序列推荐任务中,通过解耦设计,模型可以从用户行为历史数据中学习分别对动态因素和静态因素敏感的独立隐向量,一并构成了用户偏好的解耦表示,可以改善序列推荐模型的表现。学习用户隐式偏好的解耦表示十分困难,少有文献研究此问题,本文提出了一种新的变分自编码神经网络来学习解耦后的用户表示。本文的主要贡献包括:1)设计了一种全新的变分自编码框架来学习解耦后的用户偏好表示;2)使用最大均值差异(Maximum Mean Discrepancy, MMD)作为变分自编码框架中隐向量的先验分布和后验分布的正则化项;3)在标准数据集上评估了所提出的 DSVAECF 框架,实验并分析了各项参数对推荐性能的影响。

## 2 相关工作

### 2.1 基于深度学习的协同过滤推荐系统

推荐系统在过去十多年得到了广泛的研究。在协同过滤的推荐框架中,推荐问题大多被建模为预测问题:通过用户过去的选择,即用户的历史行为记录来预测用户的喜好。近年来,推荐领域的主要关注点之一是与深度学习相结合的协同过滤方法。其中,神经协同过滤(Neural Collaborative Filtering, NCF)<sup>[8]</sup>将矩阵分解扩展为非线性变换,即利用隐因子转换的简单多层感知机网络来对用户和物品的交互进行建模。GRU4Rec<sup>[9]</sup>提出了一种基于门控循环单元(Gated Recurrent Units, GRU)的循环神经网络模型,用于预测用户会话中的下一个物品。循环推荐网络(Recurrent Recommender Networks, RRN)<sup>[10]</sup>采用了长短期记忆(Long Short Term Memory, LSTM)网络构建模型,以适应用户和物品的动态性,并利用学习到的用户和物品的动态嵌入来预测用户对物品的评分。

### 2.2 基于变分自编码器的推荐系统

基于对偏好矩阵的特征进行自编码的思想,AutoRec<sup>[11]</sup>利用自编码器来编码用户的历史行为。隐式偏好可以用重构的解码来表示,重构的解码包括了所有可能感兴趣的物品的得分。变分自编码框架<sup>[12]</sup>的进一步引入表明了深度学习与隐因子模型之间的紧密联系。序列变分自编码器(Sequential Variational Autoencoders, SVAE)<sup>[13]</sup>和多项式变分自编码器(Multinomial Variational Autoencoders, MVAE)<sup>[14]</sup>分别利用时变和时不变序列来建模用户的隐式表示。SVAE结合循环神经网络的变分自编码器来对用户偏好进行建模,得到的用户偏好是一个动态变化的序列。相比之下,MVAE引入了一个具有多项式似然性的生成模型,并利用贝叶斯推理进行

参数估计,得到的用户偏好是静态的。

### 2.3 解耦表示学习

序列数据中学习到的用户静态偏好和动态偏好因素是不可互相替换的,即用户的长期静态行为并不能反映其顺序依赖的短期动态行为。解耦表示学习旨在对影响数据的关键因素进行建模,使得某一关键因素的变化仅仅引起数据在某项特征上的变化,而其他的特征不受影响。近年来,VAE已经被用于学习音频和视频的解耦表示,如文献<sup>[15]</sup>将音频和视频分解为静态部分和动态部分。在此基础上,R-WAE<sup>[16]</sup>将潜在变量的后验及其对应的先验分布的 Wasserstein 距离作为正则项,C-DSVAE<sup>[17]</sup>则通过对比学习使解耦表示与输入数据的互信息最大化,并使解耦表示间的互信息最小化。本文使用解耦表示学习方法来建模用户行为序列,以此分离用户的静态偏好和动态偏好。

## 3 基于解耦变分自编码器的协同过滤推荐算法

本节将详细介绍所提出的 DSVAECF 模型的推断过程,以及损失函数和具体的实现细节。

### 3.1 问题定义

设用户集  $U$  包含  $M$  个用户,物品集  $I$  包含  $N$  个物品。偏好矩阵为  $\mathbf{R} \in \{0, 1\}^{M \times N}$ ,  $\mathbf{R}$  中的元素  $r_{u,i} = 1$  时表示用户  $u$  和物品  $i$  有交互, $r_{u,i} = 0$  时表示用户  $u$  和物品  $i$  没有交互。

给定用户  $u$ , 定义  $I_u = \{i \in I \mid r_{u,i} = 1\}$ ,  $N_u = |I_u|$ ,  $I_u$  包含所有与用户  $u$  有交互的物品。与此同时,为偏好矩阵  $\mathbf{R}$  添加时间标签  $\mathbf{D} \in \mathbf{R}^{M \times N}$ ,  $\mathbf{D}$  中的元素  $d_{u,i}$  表示用户  $u$  和物品  $i$  交互的时间,当  $r_{u,i} = 0$  时  $d_{u,i}$  为空。令  $x_t (1 \leq t \leq N_u)$  表示按  $d_{u,i}$  先后排列的  $I_u$  的第  $t$  项, $x_{1:T}$  表示用于训练的序列  $x_1, \dots, x_T$ 。编码器的目的是利用  $x_{1:T}$  学习解耦表示  $\mathbf{z}_t = (\mathbf{z}^c, \mathbf{z}^m)$ , 其中隐向量  $\mathbf{z}^m$  与时间步  $t$  相关,隐向量  $\mathbf{z}^c$  与时间步  $t$  无关。解耦器通过隐向量  $\mathbf{z}_t$  重构出用户行为  $\hat{x}_t$ , 使其真实的用户行为尽可能一致。

### 3.2 解耦变分推断

令  $X_t, \mathbf{Z}_t, \mathbf{Z}^c$  和  $\mathbf{Z}_t^m$  分别作为  $x_t, \mathbf{z}_t, \mathbf{z}^c$  和  $\mathbf{z}_t^m$  的随机变量。为了使重构的  $X_t$  和真实的用户行为相似,假设  $\mathbf{Z}^c$  和  $\mathbf{Z}_t^m$  相互独立,则有生成模型:

$$P(X_{1:T}, \mathbf{Z}_{1:T}) = P(\mathbf{Z}^c) \prod_{t=1}^T P(\mathbf{Z}_t^m) P_\theta(x_t \mid \mathbf{Z}_t) \quad (1)$$

其中,  $P(\mathbf{Z}^c)$  和  $P(\mathbf{Z}_t^m)$  分别是  $\mathbf{Z}^c$  和  $\mathbf{Z}_t^m$  的先验分布,  $X_t$  通过条件概率分布  $P_\theta(x_t \mid \mathbf{Z}_t)$  来重构。

VAE 的任务则是从训练数据  $X_{1:T}$  中学习隐向量  $\mathbf{Z}_{1:T}$  的后验分布  $P(\mathbf{Z}_{1:T} \mid X_{1:T})$ , 但这个分布通常难以求解,因此用另一个分布  $Q(\mathbf{Z}_{1:T} \mid X_{1:T})$  去近似  $P(\mathbf{Z}_{1:T} \mid X_{1:T})$ 。变量  $\mathbf{Z}_c$  和  $\mathbf{Z}_t^m$  满足正态分布,其均值和方差通过编码器获得。为此,在式(2)中分别为  $\mathbf{Z}_c$  和  $\mathbf{Z}_t^m$  这两个变量设计了不同的编码器  $Q_\psi$  和  $Q_\varphi$ 。基于这两个编码器,  $Q(\mathbf{Z}_{1:T} \mid X_{1:T})$  可以定义为:

$$Q(\mathbf{Z}_{1:T} \mid X_{1:T}) = Q_\psi(\mathbf{Z}^c \mid X_{1:T}) \prod_{t=1}^T Q_\varphi(\mathbf{Z}_t^m \mid X_{1:t-1}) \quad (2)$$

为了使  $P(\mathbf{Z}_{1:T} \mid X_{1:T})$  和  $Q(\mathbf{Z}_{1:T} \mid X_{1:T})$  这两个分布接近,文献<sup>[18]</sup>根据变分推断(Variational Inference)理论,可以为观察到的样本推导出证据下界(Evidence Lower Bound, ELBO),如式(3)所示:

$$L(X_{1:T}) = -E_{Q_{\phi}(Z_c | X_{1:T})} \sum_{t=1}^T E_{Q_{\phi}(Z_t^m | X_{1:t})} [\log P_{\theta}(X_t | Z_t)] + \beta_1 \sum_{t=1}^T (Q_{\phi}(Z_t^m | X_{1:t-1}), P(Z_t^m)) + \beta_2 (Q_{\phi}(Z^c | X_{1:T}), P(Z^c)) \quad (3)$$

式(3)中的第一项表示 VAE 模型的重构误差,即模型的输出和输入的差异。在本文中 VAE 模型输出的是预测的用户在时间步  $t$  时刻的偏好  $\hat{x}_t$ ,即用户从时间步  $t$  到  $t+K$  可能会交互的物品,其中  $K$  为一个可自由设置的参数,表示模型需要着重预测的用户在时间步  $t$  后的  $K$  项物品。若要使模型做出准确的预测,则需要最小化  $\hat{x}_t$  和真实的用户行为之间的差异。

式(3)中的第二项和第三项为正则化项,用来防止模型为减小重构损失而退化成普通的自编码器,从而失去原始采样的能力。 $\beta_1, \beta_2$  为退火参数,当  $\beta_1, \beta_2$  小于 1 时,将削弱先验约束的影响,这意味着该模型不太可能通过原始抽样生成新的用户历史行为。但模型的最终目的是做出好的推荐,而不是产生虚构的用户历史。为此,将  $\beta_1, \beta_2$  作为一个能自由调节的正则化参数,并设它们的初始值为 0。随着梯度更新,缓慢增加  $\beta_1$  和  $\beta_2$  直到设定的值,迫使潜向量  $Z^c$  和  $Z_t^m$  向其先验分布  $P(Z^c)$  和  $P(Z_t^m)$  逼近。此过程可以看作是从普通的自编码器到变分自编码器的退火过程<sup>[19]</sup>。

### 3.3 模型架构

由图 1 所示,解耦序列变分自编码器(Disentangled Sequential Variational Autoencoder, DSVAECF)模型主要分为编码器、采样过程和解码器 3 个部份:1)编码器部分包括 RNN 编码器和 MLP 编码器,用户的行为历史序列经由两个编码器分别得到用户动态与静态偏好表示的均值和标准差;2)采样部分利用上述两个编码器输出的均值和标准差,分别得到服从高斯分布的用户偏好的动态和静态表示;3)解码器部分计算用户和目标物品表示的匹配分数,用于训练模型或进行推荐。

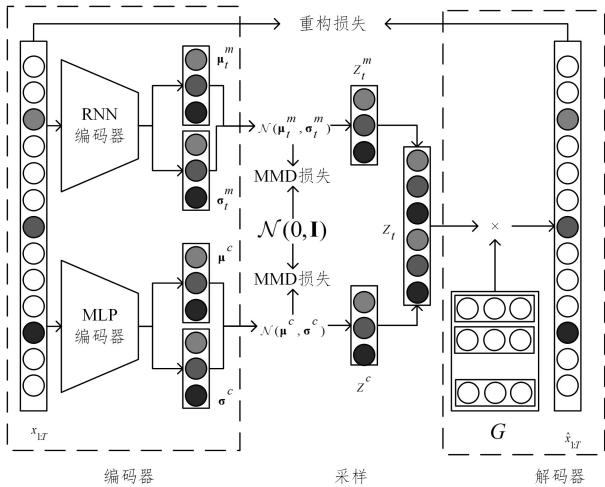


图 1 DSVAECF 模型示意图

Fig. 1 Illustration of DSVAECF

#### 3.3.1 编码器和采样过程

在训练过程中需要对随机变量的参数求导,但采样操作不可导,进而导致模型无法反向传播。因此在式(4)和式(5)中引入一个重参数化技巧<sup>[4]</sup>,从均值和标准差分别为  $\mu$  和  $\sigma$

的正态分布中采样  $Z$ ,等价于从标准正态分布  $N(0, \mathbf{I})$  中采样  $\varepsilon$ ( $\varepsilon$  的值是确定的,不参与学习),然后与  $\mu$  和  $\sigma$  做线性操作,这使得整个过程是可导的。

具体来说,式(2)中  $Q_{\phi}(Z_c | X_{1:T})$  是一个以  $\phi$  参数化并使用多层感知机(Multilayer Perceptron, MLP)建模的用户历史行为的正态分布。通过重参数技巧,  $Z^c$  通过此分布的均值  $\mu^c$  和标准差  $\sigma^c$  获得。

$$\mu^c, \ln \sigma^c = \text{MLP}(X_{1:T}) \quad (4)$$

$$Z^c = \mu^c + \sigma^c \odot \varepsilon, \varepsilon \sim N(0, \mathbf{I})$$

其中,  $\odot$  表示对应位置的元素相乘。

$Q_{\phi}(Z_t^m | X_{1:T})$  是一个以  $\phi$  参数化并使用循环神经网络(Recurrent Neural Network, RNN)建模的用户历史行为的正态分布。通过把递归层在时间步  $t$  时的输出  $H_t$  输入 MLP 中,得到此分布的均值和标准差分别为  $\mu_t^m$  和  $\sigma_t^m$ ,再通过重参数技巧得到  $Z_t^m$ :

$$H_t = \text{RNN}(H_{t-1}, X_t)$$

$$\mu_t^m, \ln \sigma_t^m = \text{MLP}(H_t) \quad (5)$$

$$Z_t^m = \mu_t^m + \sigma_t^m \varepsilon, \varepsilon \sim N(0, \mathbf{I})$$

#### 3.3.2 解码器

通过编码器得到两个不同的用户偏好的隐向量  $Z^c, Z_t^m$  后,将它们拼接为  $Z_t$ ,有  $Z_t = [Z^c, Z_t^m]$ 。在此给每个物品  $i$  定义一个目标物品嵌入  $G_i$ ,计算  $Z_t$  和  $G_i$  之间的余弦相似度并将其作为所有物品的推荐评分  $s_i$ 。

$$s_i = \frac{1}{\tau} \frac{Z_t^T G_i}{\|Z_t\|_2 \|G_i\|_2} \quad (6)$$

其中,  $\tau$  表示把相似度  $s_i$  限制在  $[-\frac{1}{\tau}, \frac{1}{\tau}]$  之间。对所有物品的评分进行 softmax 操作,得到以  $\theta$  参数化的重构的用户行为分布  $P_{\theta}(X_t | Z_t)$ ,即预测的用户在每个时间步的行为  $\hat{x}_t$ 。 $N$  表示物品的总数,则有:

$$P_{\theta}(X_t | Z_t) = \hat{x}_t = \text{softmax}([s_1; s_2 \cdots; s_N]) \quad (7)$$

#### 3.3.3 最大均值差异 MMD

式(3)中,  $D(\cdot)$  表示正则化项,用于最小化先验概率分布  $P$  与后验概率分布  $Q$  之间的差异,常使用 KL 散度,本文使用 MMD<sup>[20]</sup>。尽管两者都可以衡量两个分布之间的差异,但相比 KL 散度具有无界、不对称等缺点,式(8)中, MMD 通过把两个分布映射到再生希尔伯特空间来计算它们之间的距离,更适合用于最小化两个分布的差异<sup>[18]</sup>,使用  $T$  表示用户行为的长度,  $\hat{Z}$  表示来自  $P$  的实例,  $Z$  表示来自  $Q$  的实例,  $k(x, y) = \exp(-\frac{\|x-y\|^2}{2})$  表示高斯核函数。因为  $Z^c$  与时间步  $t$  无关,所以对于  $Z^c$  来说,  $T=1$ ,则  $Z^c$  和  $Z_t^m$  的先验分布与后验分布的 MMD 如式(9)所示。

$$\mathbb{D}(Q_{\phi}(Z_t^m | X_{1:T}), P(Z_t^m)) = \text{MMD}(Q_{\phi}(Z_t^m | X_{1:T}), P(Z_t^m))$$

$$\mathbb{D}(Q_{\phi}(Z^c | X_{1:T}), P(Z^c)) = \text{MMD}(Q_{\phi}(Z^c | X_{1:T}), P(Z^c)) \quad (8)$$

$$\text{MMD}(Q_{\phi}(Z_t^m | X_{1:T}), P(Z_t^m)) = \frac{1}{T(T-1)} \sum_{i \neq j}^T k(Z_t^m, Z_t^m) + \frac{1}{T(T-1)} \sum_{i \neq j}^T k(\hat{Z}_t^m, \hat{Z}_t^m) - \frac{2}{T^2} \sum_{i, j}^T k(Z_t^m, \hat{Z}_t^m)$$

综上所述,最终损失函数由式(3)改写为式(10),通过最小化式(10)可以学习到模型中的参数。

$$\text{MMD}(Q_\phi(\mathbf{Z}^c | X_{1:T}), P(\mathbf{Z}^c)) = -2k(\mathbf{Z}_i, \hat{\mathbf{Z}}_j) \quad (9)$$

$$L(X_{1:T}) = -\mathbb{E}_{Q_\phi(\mathbf{Z}^c | X_{1:T})} \sum_{i=1}^T \mathbb{E}_{Q_\phi(\mathbf{z}_i^m | X_{1:i-1})} [\log P_\theta(X_i | \mathbf{Z}_i)] + \beta_1 \left( \frac{1}{T(T-1)} \sum_{i \neq j}^T k(\mathbf{Z}_i^m, \mathbf{Z}_j^m) + \frac{1}{T(T-1)} \sum_{i \neq j}^T k(\hat{\mathbf{Z}}_i^m, \hat{\mathbf{Z}}_j^m) - 2 \frac{1}{T^2} \sum_{i,j}^T k(\mathbf{Z}_i^m, \hat{\mathbf{Z}}_j^m) \right) - 2\beta_2 k(\mathbf{Z}^c, \hat{\mathbf{Z}}^c) \quad (10)$$

## 4 实验设计与验证

### 4.1 数据集

在两个流行的公开数据集 MovieLens-1M 和 Amazon-video games 上评估 DSVAECF 和基准方法。这两个数据集都是时间序列数据集,包含用户-物品评分对以及相应的时间戳。由于 DSVAECF 使用的为隐式反馈,因此将两个数据集二值化,即只考虑用户提供的评分在 1~5 范围内严格大于 3 的用户-物品对。在数据预处理时,本文丢弃了 MovieLens 中少于 5 个交互记录的用户,以及 Amazon 中少于 5 个交互记录的用户和物品。在对数据集进行划分时,将 MovieLens 中 25% 的用户平分为验证集和测试集,将余下 75% 的用户作为训练集;将 Amazon 中 38% 的用户平分为验证集和测试集,将余下 62% 的用户作为训练集。表 1 列出了两个数据集过滤后的信息。

表 1 经过过滤后的数据集信息

Table 1 Information of filtered data sets

Datasets	MovieLens	Amazon
Users	6 000	19 000
Items	3 691	18 352
Interactions	950 316	174 683
Sparsity	95.701%	99.950%

### 4.2 评估标准和方法

由于 DSVAECF 模型考虑的是隐式偏好,因此评估是根据 top- $n$  推荐完成的。为了评估推荐和偏好排序的有效性,采用了 3 种常用的评估指标:召回率  $Recall@n$ 、精确率  $Precision@n$  和归一化折损累计增益  $NDCG@n$ 。

实验中,上述指标中  $n$  分别为 10 和 100。评估方案如下<sup>[19]</sup>:将验证集和测试集中的每个用户的历史行为按照时间顺序分成前 80% 和后 20% 两部分,分别作为 fold-in 和 fold-out 部分。DSVAECF 在训练时使用训练集中用户的完整历史行为进行训练和评估。在测试过程中,对于验证集和测试集中的每个用户,使用 fold-in 部分来学习必要表示的基础,并提供一个推荐列表,然后使用用户历史行为的 fold-out 部分对该推荐列表进行评估。使用按时间排序的分割方法而不是随机分割能够有效提升评估的公平性。

### 4.3 基准方法

将 DSVAECF 模型与当前最先进的各种模型进行比较。下面简要介绍本文采用的基准方法。

(1)FISM<sup>[21]</sup>是基于矩阵分解的隐因子模型,通过把用户历史行为的嵌入表示映射到隐式空间来学习物品之间的相似性,最终根据相似矩阵求得各个物品的推荐得分。

(2)MVAE<sup>[14]</sup>是采用了变分自编码器框架的推荐模型,其与 FISM 一样不考虑用户历史行为的时间顺序,而是把这些物品的嵌入表示的均值输入到解码器中。

(3)GRU4Rec<sup>[9]</sup>是基于循环神经网络 GRU 的推荐模型,该模型根据迄今为止的历史行为来预测用户会话中的下一个项目。

(4)SVAE<sup>[13]</sup>是在变分自编码器框架的基础上考虑了用户表示的时间依赖关系的模型。该模型在编码器中使用循环神经网络来捕获用户历史行为的时间依赖关系。

### 4.4 实验设置

DSVAECF 使用 PyTorch 来实现,并在单个 GTX 2080Ti GPU 上进行训练,其主要设置如下:物品的嵌入维度为 256, RNN 编码器的 GRU 循环层具有 200 个单元,MLP 编码器的输出维度和隐式空间的维度大小都为 64,正则化系数  $\beta_1$ ,  $\beta_2$  在 [0, 0.2, 0.5, 0.9] 范围内选取,训练时 Adam 优化器的学习率设置为 0.001。与文献[19]相同,每批训练数据的数目设置为 1。在训练集上训练了 25 个 epoch 之后,将在验证集上取得最好  $NDCG@100$  值的模型参数用于测试集,并将其作为最终的测试结果。

### 4.5 实验结果与分析

#### 4.5.1 推荐性能对比

为了公平地比较,实验中对不同方法的参数设置是相同的,DSVAECF 与基准方法的对比结果如表 2 和表 3 所列。

表 2 在 MovieLens 数据集上的对比实验结果

Table 2 Comparative experimental results on MovieLens dataset

Models	NDCG@		Recall@		Precision@	
	10	100	10	100	10	100
FISM	9.29	19.47	6.60	36.08	8.24	5.41
MVAE	9.84	21.38	7.35	39.85	8.24	5.63
GRU4Rec	19.49	31.58	14.33	51.11	15.82	7.19
SVAE	19.83	31.79	14.33	51.49	16.49	7.21
DSVAECF	<b>20.53</b>	<b>32.87</b>	<b>14.68</b>	<b>53.63</b>	<b>16.89</b>	<b>7.57</b>

表 3 在 Amazon 数据集上的对比实验结果

Table 3 Comparative experimental results on Amazon dataset

Models	NDCG@		Recall@		Precision@	
	10	100	10	100	10	100
FISM	0.62	1.45	0.70	4.45	0.14	0.09
MVAE	2.99	6.05	4.64	17.79	0.91	0.37
GRU4Rec	2.52	5.27	4.00	15.73	0.81	0.35
SVAE	2.10	4.07	2.97	11.49	0.57	0.26
DSVAECF	<b>3.16</b>	<b>6.44</b>	<b>4.75</b>	<b>18.57</b>	<b>0.98</b>	<b>0.40</b>

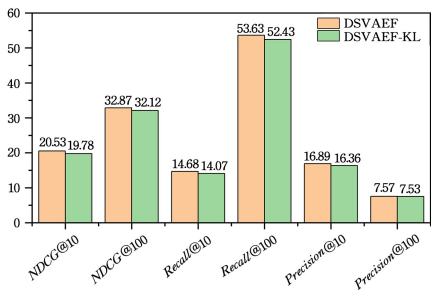
可以看到:1)在两个数据集上,DSVAECF 始终优于基准方法,在所有评估指标上都取得了最佳效果;2)DSVAECF 的表现优于 SVAE 和 MVAE 的主要原因在于,DSVAECF 使用不同的编码器来学习用户偏好的解耦表示,综合考虑了用户短期和长期的偏好因素,使推断出的用户行为和真实的用户

行为间的互信息最大化,与之相比,SVAE 和 MVAE 虽然都使用了类似的 VAE 框架,但忽视了用户偏好的耦合性,未对用户的行为历史序列进行解耦化的处理;3)DSVAECF 的表现比 GRU4Rec 好的原因可能是,GRU4Rec 是基于会话的推荐,无法通过显式地对用户的偏好建模来理解用户的一般兴趣;4)DSVAECF 的表现优于 FISM,这是因为 FISM 是基于矩阵分解的方法,所以没有考虑用户-物品交互的时序性,且 FISM 模型计算物品间的相似度只能够捕获用户的一般兴趣,而无法对用户的短期动态偏好进行建模。

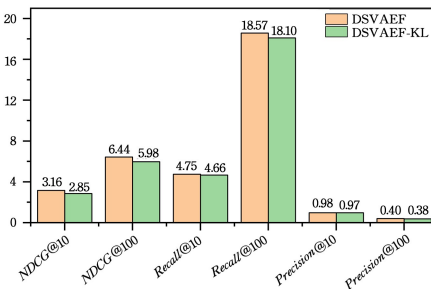
与此同时:1)在 MovieLens 数据集上的所有评估结果都优于在 Amazon 数据集上的,一个主要的原因是它们的稀疏率不同,而稀疏率与推荐性能成负相关;2)DSVAECF,SVAE 和 GRU4Rec 的推荐性能明显优于 FISM 和 MVAE,主要原因是 DSVAECF,SVAE 和 GRU4Rec 都是序列模型,能够捕获用户的短期偏好行为,而 FISM 和 MVAE 都对用户的长期行为建模,因此无法从用户的连续行为数据中准确捕获用户的真实偏好。

#### 4.5.2 采用不同分布差异的度量的表现

为了进一步分析 DSVAECF 取得优异性能的原因,本文提出了一种它的变体模型 DSVAECF-KL,并比较它们在两个推荐数据集上的性能。与 DSVAECF 相比,DSVAECF-KL 使用了 KL 散度来代替 MMD 作为正则化项。DSVAECF 和 DSVAECF-KL 的推荐性能的对比结果如图 2 所示,在两个数据集上 DSVAECF 的各个指标都优于 DSVAECF-KL。这一实验结果表明:相比 KL 散度,MMD 更适合度量两个不同但相关的分布之间的差异,使得隐向量  $z^i$  和  $z^m$  向它们的先验分布靠近,提高了隐向量的生成能力和模型的表达能力。



(a) MovieLens



(b) Amazon

图 2 在两个不同数据集上 DSVAECF 和 DSVAECF-KL 的推荐性能对比

Fig. 2 Recommended performance comparison of DSVAECF and DSVAECF-KL on two different datasets

#### 4.5.3 DSVAECF 在不同的用户历史行为长度下的表现

为了分析用于训练的用户历史行为序列长度对序列推荐模型性能的影响,通过输入不同的用户历史行为长度对模型进行训练,得到  $NDCG@100$  的值,如表 4 和表 5 所列。从表中可以看到,虽然在 MovieLens 上 MVAE 和 SVAE 分别在用户历史行为序列长度为 [1-10] 和 [101-150] 时取得了最优结果,但在其他情况下都不如 DSVAECF。而在 Amazon 数据集上,无论用户历史行为序列长度为多少,DSVAECF 都取得了最佳的结果。此外,在历史行为长度超过 50 以后 3 个模型的性能都随着序列的增加而降低。

表 4 在 MovieLens 数据集上用户历史行为长度对  $NDCG@100$  的影响

Table 4 Impact of user's historical behavior length on  $NDCG@100$  on MovieLens dataset

(单位:%)

Models	[1, 10]	[11, 20]	[21, 50]	[51, 100]	[101, 150]	[151, 200]	[201, 300]	[301, -]
MVAE	14.24	21.06	24.20	19.52	21.31	20.70	18.53	18.51
SVAE	6.56	24.37	35.60	32.27	<b>32.14</b>	26.60	26.13	21.22
DSVAECF	7.59	<b>24.95</b>	<b>36.58</b>	<b>33.06</b>	31.46	27.13	27.13	<b>26.78</b>

表 5 在 Amazon 数据集上用户历史行为长度对  $NDCG@100$  的影响

Table 5 Impact of user's historical behavior length on  $NDCG@100$  on Amazon dataset

(单位:%)

Models	[1, 10]	[11, 20]	[21, 50]	[51, 100]	[101, 150]	[151, 200]	[201, 300]	[301, -]
MVAE	2.67	3.61	7.42	7.22	1.11	1.59	1.60	-
SVAE	3.41	3.00	6.16	4.82	6.72	7.32	7.05	-
DSVAECF	7.03	<b>8.10</b>	<b>10.61</b>	<b>12.26</b>	<b>10.24</b>	<b>9.66</b>	<b>8.79</b>	-

#### 4.5.4 退火参数对推荐性能的影响

DSVAECF 模型在 MMD 正则化损失部分设计了退火机制,以权衡正则项对推荐效果的影响。本节通过网格搜索策略分析两个参数值的不同组合对  $NDCG@100$  值的影响,结果如表 6 所列。可以看出:1)当  $\beta_1=0, \beta_2=0$  时  $NDCG@100$  最小,即当完全去掉正则化项,使模型不具有原始抽样能力时模型的推荐效果最差;2)当  $\beta_1$  和  $\beta_2$  其中之一为 0 时,无论另一个参数取值如何, $NDCG@100$  都比二者同时为 0 时高,这充分说明了保留一定的原始抽样能力可以提高模型的表达能力,有利于提高推荐效果。

表 6 MovieLens 数据集上 DSVAECF 模型的两个退火参数对  $NDCG@100$  的影响

Table 6 Impact of two annealing parameters of DSVAECF model on  $NDCG@100$  on MovieLens dataset

(单位:%)

$\beta_2$	$\beta_1=0$	$\beta_1=0.2$	$\beta_1=0.5$	$\beta_1=0.9$
0	31.09	31.71	31.61	32.04
0.2	31.60	32.04	31.63	31.68
0.5	31.79	32.00	32.01	31.84
0.9	31.75	32.04	32.04	32.02

#### 4.5.5 K 值对推荐性能的影响

大多数生成模型仅预测用户的下一项,为了提高模型的灵活性,DSVAECF 模型的预测扩展到了后面的第  $K$  项。图 3 给出了这种能力的分析结果。从图中可以看出,在 MovieLens 上,当  $K=4$  时  $NDCG@100$  取得最佳值, $K$  的可接受值的范围为  $1\sim 10$ ,随着  $K$  增大,推荐性能逐渐下降。而在 Amazon 数据集上当  $K=1$  时  $NDCG@100$  取得最佳值,随着  $K$  增大推荐性能逐渐下降,然后趋于平缓。

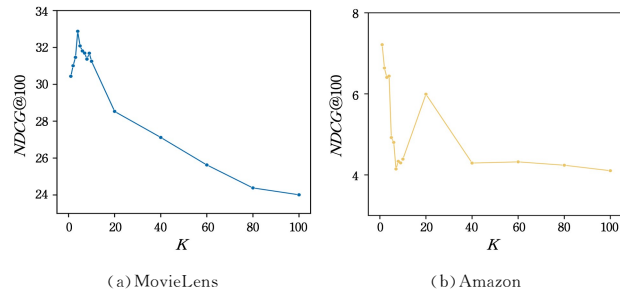


图 3 在两个数据集上 DSVAECF 的不同  $K$  值对  $NDCG@100$  的影响

Fig. 3 Impact of different  $K$  values of DSVAECF on  $NDCG@100$  on two datasets

#### 4.5.6 隐向量维度对推荐性能的影响

考虑到 DSVAECF 使用隐向量来表达用户偏好,本节分析了不同隐向量维度对 DSVAECF 推荐性能的影响。在 MovieLens 及 Amazon 数据集上,隐向量维度参数的实验结果如图 4 所示。由图可知,在 MovieLens 上不同的隐向量维数对  $NDCG@100$  的影响并不显著,而在 Amazon 数据集上隐向量维度为 30 时  $NDCG@100$  取得最佳,在隐向量维度为 100 时有一个回升,随着维度的增加总体趋势趋于下降。

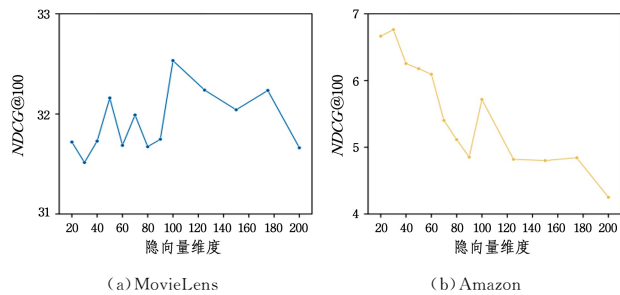
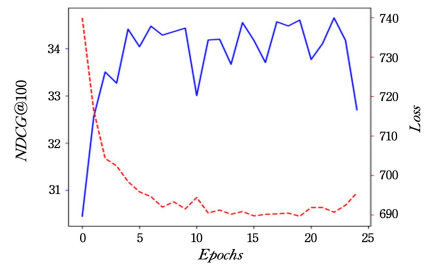


图 4 在两个数据集上 DSVAECF 不同隐向量维度对  $NDCG@100$  的影响

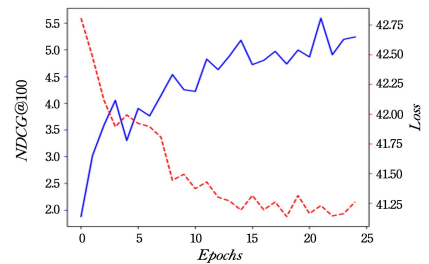
Fig. 4 Effects of different latent vector dimensions of DSVAECF on  $NDCG@100$  on two datasets

#### 4.5.7 训练收敛曲线

本节还分析了 DSVAECF 的收敛速度。根据图 5 中的  $NDCG@100$  值(左  $y$  轴和蓝线)和损失值(右  $y$  轴和红线)可知,模型在学习阶段收敛的速度很快,未表现出过拟合的特征。特别地,模型在 MovieLens 上经过 10 个 epoch 左右开始收敛,而在 Amazon 上需要 20 个 epoch 左右。实验中,在 MovieLens 上每个 epoch 的平均运行时间是 125 s,而在 Amazon 上是 30 min,学习速度与数据集中的交互数量呈线性关系。



(a) MovieLens



(b) Amazon

图 5 DSVAECF 在两个数据集上的学习曲线(电子版为彩图)

Fig. 5 Learning curve of DSVAECF on two datasets

**结束语** 将变分自编码器提供的隐式空间表示能力与循环神经网络的序列建模能力相结合,是一种有效的序列推荐策略。在此基础上,本文设计了 DSVAECF 模型来解决用户偏好因素的耦合性问题。实验结果表明,DSVAECF 通过不同的编码器来分别捕获用户的静态和动态偏好因素,以此得到用户偏好的解耦表示,提升了模型的推荐性能。

DSVAECF 分别使用了与 SVAE 和 MVAE 相同的 RNN 和 MLP 编码器,从而体现解耦学习的优越性。在未来的工作中,将对编码器做进一步的扩展,同时在学习用户的解耦隐向量的过程中增加其他的辅助策略。

## 参考文献

- [1] CHAE D K, KANG J S, KIM S W, et al. Cfgan: A generic collaborative filtering framework based on generative adversarial networks[C]//Proceedings of the 27th ACM International Conference on Information and Knowledge Management. 2018:137-146.
- [2] KOREN Y, BELL R, VOLINSKY C. Matrix factorization techniques for recommender systems[J]. Computer, 2009, 42(8): 30-37.
- [3] ZHANG S, YAO L, SUN A, et al. Deep learning based recommender system: A survey and new perspectives[J]. ACM Computing Surveys(CSUR), 2019, 52(1): 1-38.
- [4] KINGMA D P, WELING M. Auto-Encoding Variational Bayes [J]. arXiv:1312.6114, 2013.
- [5] REZENDE D J, MOHAMED S, WIERSTRA D. Stochastic Backpropagation and Approximate Inference in Deep Generative Models[C]// Proceedings of the International Conference on Machine Learning. PMLR, 2014:1278-1286.
- [6] WANG S J, HU L, WANG Y, et al. Sequential Recommender Systems: Challenges, Progress and Prospects[C]// Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence(IJCAI-19). 2019.

- [7] ZHANG Y, ZHU Z W, HE Y, et al. Content-Collaborative Disentangled representations Learning for Enhanced Recommendation[C]// Proceedings of the Fourteenth ACM Conference on Recommender Systems. 2020:43-52.
- [8] HE X N, LIAO L Z, ZHANG H W, et al. Neural Collaborative Filtering[C]// Proceedings of the 26th International Conference on World Wide Web. 2017:173-182.
- [9] HIDASI B, KARATZOGLOU A, BALTRUNAS L, et al. Session-based Recommendations with Recurrent Neural Networks [J]. arXiv:1511.06939, 2018.
- [10] WU C Y, AHMED A, BEUTEL A, et al. Recurrent recommender networks[C]// Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. 2017:495-503.
- [11] SEDHAIN S, MENON A K, SANNER S, et al. AutoRec: Autoencoders Meet Collaborative Filtering[C]// Proceedings of the 24th International Conference on World Wide Web. 2015:111-112.
- [12] SHENBIN I, ALEKSEEV A, TUTUBALINA E, et al. RecVAE: A new variational autoencoder for Top-N recommendations with implicit feedback[C]// Proceedings of the 13th International Conference on Web Search and Data Mining. 2020:528-536.
- [13] SACHDEVA N, MANCO G, RITACCO E, et al. Sequential variational autoencoders for collaborative filtering[C]// Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. 2019:600-608.
- [14] LIANG D W, KRISHNAN R G, HOFFMAN M D, et al. Variational autoencoders for collaborative filtering[C]// Proceedings of the 2018 World wide Web Conference. 2018:689-698.
- [15] LI Y Z, MANDT S. Disentangled sequential autoencoder[C]// Proceedings of the 35th International Conference on Machine Learning. 2018:5656-5665.
- [16] HAN J, MIN M R, HAN L, et al. Disentangled recurrent wasserstein autoencoder[C]// Proceedings of the 2021 International Conference on Learning Representations. 2021.
- [17] BAI J W, WANG W R, CARLA P G. Contrastively disentangled sequential variational autoencoder[C]// Proceedings of the 2021 Annual Conference on Neural Information Processing Systems. 2021.
- [18] HAN J, MIN M R, HAN L G, et al. Disentangled Recurrent Wasserstein Autoencoder[C]// Proceedings of the 9th International Conference on Learning Representations. 2020.
- [19] BOWMAN S R, VILNIS L, VINYALS O, et al. Generating sentences from a continuous space[C]// Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL. Association for Computational Linguistics (ACL), 2016:10-21.
- [20] TOLSTIKHIN I, BOUSQUET O, GELLY S, et al. Wasserstein auto-encoders[C]// Proceedings of the 6th International Conference on Learning Representations (ICLR 2018). OpenReview.net, 2018.
- [21] KABBUR S, NING X, KARYPIS G. FISM: Factored Item Similarity Models for top-N Recommender Systems[C]// Proceedings of the 19th ACM SIGKDD International. 2013:659-667.



**WU Mei-lin**, born in 1997, postgraduate. His main research interests include recommendation system and so on.



**HUANG Jia-jin**, born in 1977, Ph.D. His main research interests include recommendation system and so on.

(责任编辑:喻黎)