

# 用于多行为推荐的邻域增强的变分自编码器

杜炜豪

## 摘要

近年来,随着互联网时代的来临,每个人可接触到的信息是海量的,这对于信息消费者、信息生产者和平台管理者都充满挑战性。为了解决信息过载的问题,智能推荐技术应运而生,推荐算法可以帮助用户挑选出他们可能感兴趣的商品或内容。推荐算法建立在用户的历史反馈之上,用户反馈往往是有多种类型的。多行为推荐算法在建模过程中使用了多种类型的用户反馈来对目标反馈进行偏好预测,在该领域一个较大的挑战是不同反馈之间的异构性,如何有效结合多种反馈信息进行建模是一个问题。基于邻域的单行为推荐算法较为常见,使用单一反馈构建邻域并进行利用,而现有的多行为推荐算法中缺少利用多种反馈构建邻域模型。本文提出了一种新的结合多种反馈信息进行建模的视角,通过多种用户反馈来计算用户间的相似度并构建用户之间的邻域,再将这些邻域信息进行转化最后纳入到建模中。近年来,基于神经网络技术的协同过滤模型快速发展,其中变分自编码器的表现非常出色,本文基于变分自编码器提出了邻域增强的变分自编码器(Variational Auto-Encoder with Neighborhood, VAEN),在公开数据集上的实验结果表明本文提出的多行为推荐算法优于其他同类型的模型。

**关键词:** 推荐系统; 多行为推荐; 邻域; 变分自编码器

## 1 引言

随着互联网时代的来临,每个人可接触到的信息是海量的,这无论对信息消费者、信息生产者还是管理者而言都是充满挑战性的。基于以上处境,推荐算法应运而生。推荐系统的基本任务是联系用户和商品,可以减轻用户面临的信息过载问题以及提高用户满意度<sup>[1]</sup>。推荐算法的研究起源于20世纪90年代<sup>[2][3]</sup>,发展至今已在工业界广泛应用,各互联网公司的发展已经离不开推荐算法的应用。推荐算法建立在用户历史反馈之上,用户反馈一定程度上可以视为用户偏好的表现,通过对收集来的用户反馈进行分析建模,就能进一步对用户进行个性化推荐。用户对物品的反馈往往是多方面的,比如浏览、购买、分享、加入购物车等。

学术界对个性化推荐的研究一直在推进,并逐渐发展出多个分支。其中一种分类方式是将研究方向分为用户对物品的评分预测和推荐物品排序问题,另外对于用户的历史反馈也可以分为显式反馈和隐式反馈。显式反馈指的是能明确反映用户对物品喜欢或不喜欢的程度的指标,一般是评分数据;隐式反馈指的是不能明确反映用户对于物品喜爱或厌恶的程度,这种类型的数据一般只能将所有行为全部默认为正反馈,虽然隐式反馈存在不少弊端,包含较多噪声,但是它广泛存在,比如一名用户在某次选购过程中,浏览了物品A,将物品B加入到了购物车中,分享了物品C,但是我们无法通过这些行为衡量用户对这些物品的偏好值。评分预测算法一般基于显式反馈<sup>[4][5]</sup>,推荐物品排序算法一般基于隐式反馈<sup>[6]</sup>。由于隐式反馈的普遍性,本文将重点研究对于隐式反馈建模的推荐算法。

在推荐算法发展的早期阶段,许多算法研究都是基于单个种类的用户反馈进行的,通常被称为单类协同过滤(One-Class Collaborative Filtering, OCCF)问题<sup>[6][7][8]</sup>。由于仅有一种反馈,用户和物品之间的交互通常比较稀疏,算法效果受到一定限制。后来,将多种反馈一起纳入到建模过程中的推荐算法

被提出, 这类方法被称为异构单类协同过滤 (Heterogeneous One-Class Collaborative Filtering, HOCCF) 方法<sup>[9][10]</sup>, 又被称为多行为推荐算法, 这类方法除了考虑目标反馈外, 将其他辅助反馈也一起加入到建模过程中, 相当于通过多种反馈行为来推测一种行为发生的概率, 例如购买是目标反馈行为, 在建模时, 除了购买行为外, 还将浏览、分享等其他辅助反馈一起考虑。辅助反馈的加入可以帮助更好地推测用户的偏好。

随着近年来的发展, 多行为推荐相关的工作已经有很多, 其中大致可以分为三个类别<sup>[11]</sup>: 基于分解的方法<sup>[12][13]</sup>, 基于迁移学习的方法<sup>[14][15]</sup>和基于深度学习的方法<sup>[16][17][18]</sup>。前两个分类的方法通常基于比较传统的推荐算法, 这些方法往往是简单的线性模型, 对于比较复杂的场景可能无法有很好的效果。随着深度学习技术近年来的快速发展, 基于深度学习的推荐算法的研究也在不断推进, 深度学习可以通过更加复杂的非线性模型来对反馈行为进行建模, 使得其能应对更加复杂的场景。目前基于深度学习的多行为推荐算法的研究中, 又包括基于变分自编码器 (Variational Autoencoder, VAE)<sup>[19]</sup>和基于图神经网络 (Graph Neural Networks, GNN) 的方法。其中用于协同过滤的基于多项式分布的变分自编码器 (Multi-VAE)<sup>[19]</sup>于 2018 年被提出, 在物品推荐任务上展示出出色的效果。在 2021, 面向异构单类协同过滤的阶段式变分自编码器 (Staged variational autoencoder, SVAE)<sup>[16]</sup>被提出; 在 2022, 用于多行为推荐的 VAE++<sup>[20]</sup>被提出。这两个基于变分自编码器的推荐算法都是使用了购买和点击两种反馈行为来进行建模。购买作为目标行为数据比较稀疏, 利用数据更加丰富的点击行为作为辅助反馈帮助模型更好地捕捉用户的偏好。由于变分自编码器结构轻量化, 在物品推荐任务上效果出色, 受这两个模型启发, 本文继续基于变分自编码器和这两种反馈尝试提出效果更好的多行为推荐算法。

在多行为推荐算法领域, 一个较大的挑战是不同行为之间的异构性, 如何有效结合多种反馈信息进行建模是一个问题。经过调研发现, 在单一行为的推荐算法中, 利用目标反馈构建邻域信息的模型较为常见, 而现有的多行为推荐算法中缺少利用多种行为构建的邻域信息的模型。本文提出了一种新的利用多种反馈信息进行建模的视角, 通过多种用户反馈来计算用户间的相似度并构建用户之间的邻域信息, 再将这些信息纳入到建模中, 理论上由多种行为构建的邻域会比单一行为的更加精准和有效。最终, 将上述多行为邻域模块和 Multi-VAE 进行结合, 提出了邻域增强的变分自编码器 (Variational Auto-Encoder with Neighborhood, VAEN)。在公开数据集上的实验结果表明本文提出的多行为推荐算法优于其他同类型的模型, 验证了模型的有效性。

## 2 相关工作

### 2.1 基于自编码器的协同过滤模型

近年来, 深度学习技术在推荐系统领域得到了广泛的应用。本文主要关注的是和变分自编码器相关的自编码器模型。基于自编码器的协同过滤推荐模型 (Autoencoder-based Collaborative Filtering Recommendation Model, AutoRec)<sup>[21]</sup>是比较基础的模型自编码器模型, 被用于评分预测任务。该模型以用户对物品的评分作为输入, 映射到隐藏层中, 再经过解码器输出预测的评分。协同降噪自编码器 (Collaborative Denoising Autoencoder, CDAE)<sup>[22]</sup>在 AutoRec 的基础上作了进一步的改进, 首先对输入样本进行了降噪操作, 且将用户结点引入到模型中, 一定程度上提升了模型效果。但是在该模型中作者只探究了高斯和逻辑似然损失函数, 而事实上在推荐任务中, 使用多项式似然的自编码器模型的鲁

棒性大大增强。

## 2.2 基于变分自编码器的多行为推荐模型

阶段式变分自编码器 (Staged variational autoencoder, SVAE)<sup>[16]</sup>是一个基于迁移学习的模型。考虑从辅助反馈中提取信息来辅助目标反馈的偏好学习,一个直观的做法就是先对辅助反馈进行建模提取特征,进而将其迁移到目标反馈的学习过程中进行利用。基于这个想法, SVAE 分两个阶段由两个独立的 VAE 结构分别对辅助反馈和目标反馈进行建模。在点击阶段,使用标准的 Multi-VAE 结构对点击数据进行训练。在购买阶段,将点击阶段的隐藏向量迁移,拼接到购买数据的 Multi-VAE 结构中。

不同于 SVAE 的阶段式训练, VAE++<sup>[20]</sup>只使用一个网络结构共同学习了购买和点击两种反馈。VAE++ 基于标准的 Multi-VAE 结构,提出了两个额外的模块来对点击数据进行利用,分别是 Target Representation Enhancement module 和 Target Representation Refinement module。在 enhancement 模块中,将多种反馈的混合数据经过一个多层感知机后获得的均值向量接入到主体模型中,和购买数据的均值向量通过一个门控网络进行整合,丰富了可学习的知识空间。在 refinement 模块中,点击数据经过一个多层感知机后获得的均值向量和 Multi-VAE 的隐藏向量拼接起来后输入到解码器中。引入点击数据后,模型可以学习到购买和点击偏好之间的差异,因此用户购买表征可以被更好地建模。

## 2.3 基于邻域的推荐算法

当今基于邻域的推荐算法已经有不少工作,其主要原理是通过已有的反馈信息构建用户或者物品之间的相关性并进行利用,其中的方法主要可以分为两个类别:面向用户<sup>[23]</sup>和面向物品<sup>[24]</sup>的方法。

面向用户的方法重点关注的是具有相似历史反馈的用户,通过已有的历史反馈来计算用户之间的相似度,相似度高的用户可能有着共同的偏好,比如邻居曾经购买的物品,很可能该用户也会购买,通过算法来学习到他们之间的一些共性,可以更好地捕捉用户的偏好以进行个性化推荐。

面向物品的方法重点关注的是物品之间的相似度,利用物品相似度来推测用户偏好,当一个物品和用户已有反馈的物品的相似度高,那么很可能这个物品也是该用户偏好的。

本文研究的是面向用户的邻域方法。对于隐式反馈,常见的计算用户间相似度的方法有两个,分别是余弦相似度和杰卡德指数 (Jaccard Index),使用  $\text{Sim}_{uw}$  表示用户  $u$  和用户  $w$  之间的相似度,  $I_u$  和  $I_w$  分别指的是两个用户有过历史反馈的物品集,具体计算方式如下:

### 2.3.1 余弦相似度

$$\text{Sim}_{uw} = \frac{|I_u \cap I_w|}{\sqrt{|I_u|}\sqrt{|I_w|}} \quad (1)$$

### 2.3.2 杰卡德指数

$$\text{Sim}_{uw} = \frac{|I_u \cap I_w|}{|I_u \cup I_w|} \quad (2)$$

下面是一个较为基础的面向用户的邻域方法利用用户相似度预测用户  $u$  对物品  $i$  的偏好的例子：

$$\hat{r}_{ui} = \sum_{w \subseteq U_i \cap S_u} \text{Sim}_{uw} \quad (3)$$

其中， $\text{Sim}_{uw}$  表示用户  $u$  和用户  $w$  之间的相似度， $U_i$  表示和物品  $i$  有过交互的用户集合， $S_u$  表示用户  $u$  最相似的邻居集合。

### 3 本文方法

近年来深度学习技术发展迅速,其在推荐算法领域的应用已经全面展开。变分自编码器 (Variational Auto-Encoder, VAE) 是一类重要的基于神经网络的生成模型, 于 2013 年被提出<sup>[25]</sup>。在 2018 年, 变分自编码器被应用到推荐算法的单类协同过滤问题上<sup>[19]</sup>, 基于多项式分布的变分自编码器 (Multi-VAE) 被提出, 并取得了出色的效果, 本文将基于 Multi-VAE 模型进行复现并作进一步的改进。下面基于单类协同过滤问题详细介绍一下 Multi-VAE 的具体结构。如图 1 所示为 Multi-VAE 的整体结构。

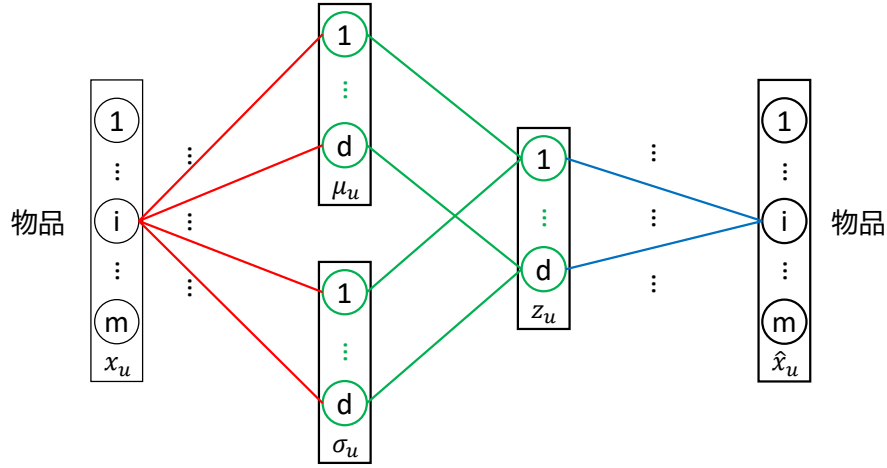


图 1: Multi-VAE 的整体结构

基于多项式分布的变分自编码器有以下假设：对于每个用户  $u$ ,  $z_u$  服从标准高斯分布,  $x_u$  服从多项式分布。

首先是输入层。将单个用户的历史反馈转化为一个由 0、1 组成的多热 (multi-hot) 向量  $x_u$ , 长度为物品集大小  $m$ , 在用户有目标反馈的物品的相应位置为 1, 其余位置为 0。

然后是编码器模块, 目标是由输入向量生成潜在特征向量  $z_u$  的分布, 输出为两个  $d$  维的向量, 分别是表示均值的  $\mu_u$  和表示标准差的  $\sigma_u$  向量。具体的计算方法如下：

$$\mu_u = f(x_u W_\mu + b_\mu) \quad (4)$$

$$\sigma_u = \exp^{f(x_u W_\sigma + b_\sigma)} \quad (5)$$

其中,  $W_\mu, W_\sigma \in R^{m \times d}$  和  $b_\mu, b_\sigma \in R^{1 \times d}$  分别是要学习的网络权重矩阵和偏置向量。

接下来是从参数  $\phi$  下的变分分布采样获得潜在特征向量  $z_u$ , 这里是一个对角高斯分布,

$$q_\phi(z_u | x_u) = N(\mu_u, \text{diag}(\sigma_u^2)) \quad (6)$$

在训练过程中, 由于采样过程是随机的, 因此很难获得梯度值, 为了在反向传播的过程中更新参数, 这里需要运用重参数技巧<sup>[25][26]</sup>。具体的做法是通过标准分布  $\varepsilon \sim N(0, \text{diag}(1))$  来近似获得潜在特

征向量  $z_u$ 。

$$z_u = \mu_u + \varepsilon \otimes \sigma_u \quad (7)$$

另外,  $z_u$  还需要通过  $q_\phi(z_u | x_u)$  和  $p(z_u)$  之间的 Kullback-Leibler (KL) 散度来正则化:

$$\mathcal{L}_{KL}(z_u) = KL(q_\phi(z_u | x_u) \| p(z_u)) \quad (8)$$

最小化  $KL(q_\phi(z_u | x_u) \| p(z_u))$  令网络学习得到的后验分布  $q_\phi(z_u | x_u)$  接近假设的先验分布  $p(z_u)$ , 一般用的是标准高斯分布。

最后是解码器模块, 目标是生成用户的历史反馈的概率分布。使用上面获得的潜在特征向量  $z_u$  作为输入, 经过一个 softmax 函数输出整个物品集上的概率分布  $\pi(z_u)$ 。给定用户  $u$  历史目标反馈的总数, 历史反馈数据向量  $x_u$  被假设可以从概率为  $\pi(z_u)$  的多项式分布中采样获得。

$$\pi(z_u) = \text{softmax}(f_\theta(z_u)) \quad (9)$$

$$x_u \sim \text{Multi}(P_u, \pi(z_u)) \quad (10)$$

其中,  $f_\theta(\cdot)$  是一个参数为  $\theta$  的多层感知机,  $\pi(z_u)$  是  $f_\theta(\cdot)$  的分布函数,  $P_u$  是用户  $u$  所有目标反馈的总数。

用  $\hat{x}_u$  表示最终重构的目标反馈向量, 理论上  $\hat{x}_u$  和输入向量  $x_u$  越相似越好, 因此有以下的重构损失:

$$\mathcal{L}(\hat{x}_u, x_u) = E_{q_\phi(z_u|x_u)} [\log p_\theta(x_u | z_u)] \quad (11)$$

因此, 总的损失函数如下:

$$\mathcal{L}_{VAE} = \mathcal{L}(\hat{x}_u, x_u) - \beta \mathcal{L}_{KL}(z_u) \quad (12)$$

其中,  $\beta \in [0, 1]$  是正则项的权重参数。

对于每个用户, 输入历史反馈数据的多热向量到变分自编码器网络中, 从最终输出的偏好预测向量中过滤掉已交互过的物品, 在其余物品中选出评分最高的  $N$  个物品作为推荐的物品。

## 4 复现与改进

### 4.1 与已有开源代码对比

使用了他人实现的 Multi-VAE 的代码, 在此基础上创新地提出了使用多种反馈构建的邻域模块, 并与 Multi-VAE 进行了结合, 提出邻域增强的变分自编码器 (Variational Auto-Encoder with Neighborhood, VAEN)。下面一节将详细介绍尝试的多种创新结构。

### 4.2 创新模型结构

#### 4.2.1 问题概述

在本文我们研究的是多行为推荐算法, 利用用户的多种历史反馈学习用户的偏好, 对目标反馈进行预测并完成物品推荐。本文中我们仅在邻域的构建上使用多行为信息, 并将邻域信息加以应用, 主体部分的变分自编码器还是仅使用单类反馈, 即目标反馈。于主体的变分自编码器而言这个邻域信息模块是额外增添的一个模块, 间接利用了额外的辅助反馈信息更好地对目标反馈进行预测。在网络训练前需要预先通过具体规则计算好各个用户的邻居, 网络的输入为用户的历史目标反馈数据以及邻居

数据，输出为预测的目标反馈数据。

#### 4.2.2 多种反馈下邻域的构建

在基于邻域方法中，对邻域的构建是最基本同时也是很关键的一个步骤，邻居能否有效选取对于结果的影响很大，相关性高的邻居有利于网络更好地捕捉到潜在的特征，进而提升模型效果；而若选取的邻居相关性不大则可能带来更多混乱的噪声信息，导致模型效果变差。构建邻域的核心步骤是计算用户间的相似度，因为邻居的选取的主要标准即用户间相似度。在有多种类型反馈的情况下，用户物品集的划分方法可以有多种，如分行为类型划分各自的物品集或者是取它们的并集，而对应地也需要采取不同的相似度计算方法。这个小节将详细介绍两种在多种类型反馈下邻域的构建方法，分别是：分反馈类型单独划分物品集的情况和对多种类型反馈的物品集取并集的情况。

##### 4.2.2.1 分反馈类型单独划分物品集

每个用户根据不同类型的反馈划分单独的物品集。本文的研究基于购买和点击两种反馈，则每个用户都有购买物品集和点击物品集两个物品集，具体表示如下：

$$I_u = \{I_u^P, I_u^\varepsilon\} \quad (13)$$

其中， $I_u^P$  表示的是用户  $u$  有过目标反馈（购买）的物品集合， $I_u^\varepsilon$  表示用户  $u$  有过辅助反馈（点击）的物品集合。在计算相似度时，两种行为分别计算相似度，然后加权相加作为最终的相似度。具体表示如下：

$$\text{Sim}_{uw} = \alpha * \text{Sim}_{uw}^P + (1 - \alpha) * \text{Sim}_{uw}^\varepsilon \quad (14)$$

其中， $\text{Sim}_{uw}$  表示用户  $u$  和用户  $w$  之间的相似度， $\text{Sim}_{uw}^P$  和  $\text{Sim}_{uw}^\varepsilon$  分别表示购买行为和点击行为下两个用户之间的相似度， $\alpha \in [0, 1]$  是衡量购买和点击两种行为相似度重要性的权重参数。当  $\alpha = 1$  时，表示仅使用购买行为计算相似度，当  $\alpha = 0$  时，表示仅使用点击行为计算相似度。

对不同的行为类型单独进行相似度计算是最直观的方法，因为理论上不同的行为对于用户偏好程度的表现是有差别的，比如购买行为一般比点击行为所表现出的偏好会更加强烈，所以分行为类型单独计算再加权相加可以保留不同行为之间的差异性，还可以通过权重系数的调整来找到合适的比例。

##### 4.2.2.2 对多种类型反馈的物品集取并集

在计算各个用户的相似度前，先将每个用户多种行为下交互过的物品取并集，当作该用户交互过的总物品集。本文的研究基于购买和点击两种反馈，具体表示如下：

$$I_u = I_u^P \cup I_u^\varepsilon \quad (15)$$

取并集后每个用户只有一个物品集，直接计算用户间的相似度即可。

对不同的行为类型取并集再进行相似度计算相当于将不同行为表示的用户偏好程度视为同等水平，如购买过或点击过的物品都一视同仁，都仅仅归类为交互过的物品。虽然这种计算方式理论上会比单独计算的方式忽略了一部分有关行为偏好强度的信息，但在一些数据缺失的情况下会比较适用，如每个用户都有一定的购买数据，但是部分用户在点击行为上数据缺失，此时如果使用单独计算的方式就无法获得这部分用户在点击行为上的相似度，可能导致最终选取的邻居不够精确，这时使用取并集的计算方式可能效果会更好。

#### 4.2.2.3 根据用户间相似度获取邻居集合

总的来说，还是需要根据数据集的实际情况来选取合适的相似度计算方式。

通过以上计算方法获得各用户之间的相似度后，对于单个用户  $u$ ，对他和其他用户的相似度排序并取相似度最高的  $K$  个用户作为其邻居集合。

至此，各用户的邻域构建完成，接下来需要对邻居信息进行转化和利用。

#### 4.2.3 邻域信息的转化

通过利用多种类型的反馈信息构建好用户间的邻域后，如何有效利用这些邻居信息是一个问题。上一步我们通过一定的计算规则获得了每个用户最相似的一部分邻居，还需要进一步将这些邻居信息转化为可用的信息向量，以便接入主体网络中。本小节将详细介绍两种将邻域信息通过神经网络技术转化为可用的信息向量，进而接入到主体模型上的方式，分别是：多热向量接多层感知机的方式和潜在特征方式。

##### 4.2.3.1 多热向量接多层感知机

对于每个用户，将选定的  $K$  个邻居转化为一个长度为  $n$  的多热向量  $s_u$ ，和变分自编码器的输入形式类似，即该用户的邻居对应位置上的值为 1，其余位置为 0，用该向量来表示用户的邻居有哪些。然后接一个多层感知机对邻居信息进行转化，获得一个  $d_v$  维的信息向量，结构如图 2 所示。具体表示如下：

$$v_u = f(s_u W_s + b_s) \quad (16)$$

其中， $v_u \in R^{1 \times d_v}$  表示转化获得的邻居信息向量， $s_u \in \{0, 1\}^{1 \times n}$  表示输入的邻居多热向量。

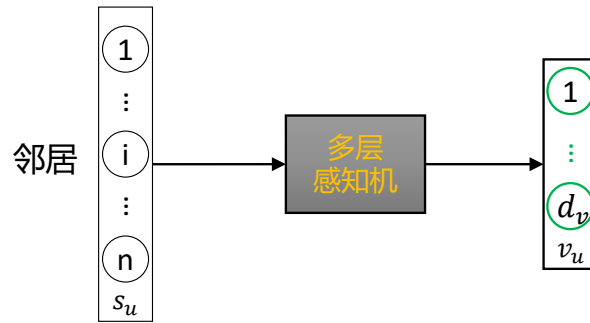


图 2: 多热向量接多层感知机转化邻居信息的结构

这是一种比较直观的转化方式，结构简单，直接通过一个多层感知机来转化邻居信息，但是可能转化信息的能力比较有限。

##### 4.2.3.2 潜在特征

用潜在特征的方式来表征邻居信息，由于模型后续可能要投稿，出于成果保护的目的，在此不详细介绍。

#### 4.2.4 在变分自编码器中的接入方式

邻居信息转化完成后，需要接入到主体模型进行应用，不同的接入方式会对最终结果产生不同的效果，因为额外信息的接入会直接影响主体模型的学习过程。在本文中我们采用基于多项式分布的变分自编码器（Multi-VAE）作为主体模型，提出邻域增强的变分自编码器（Variational Auto-Encoder with



Neighborhood, VAEN)。本小节将详细介绍两种将邻域信息向量接入到变分自编码器的方式，分别是：在输入端拼接和与  $z$  向量拼接。

#### 4.2.4.1 输入端拼接

这种方式仅针对使用多热向量作为邻居数据输入方式的情况，将邻居数据输入向量  $s_u \in \{0, 1\}^{1 \times n}$  和变分自编码器原本的输入  $x_u \in \{0, 1\}^{1 \times m}$  进行拼接，获得一个长度为  $m+n$  的输入向量，这是最直接的接入方式，并没有对邻居数据作其他转化处理和新增其他额外的模块，将这种结构命名为 VAEN(I)，结构如图 3 所示。具体表示如下：

$$x'_u = [x_u, s_u] \quad (17)$$

其中， $x'_u \in \{0, 1\}^{1 \times (m+n)}$  表示拼接后的输入向量。

这种方式将所有原始的信息都集中在一起输入到网络中，包括反馈数据和邻居数据，保留了变分自编码器的原始结构，将对邻居信息的利用融入到原本的学习过程中。

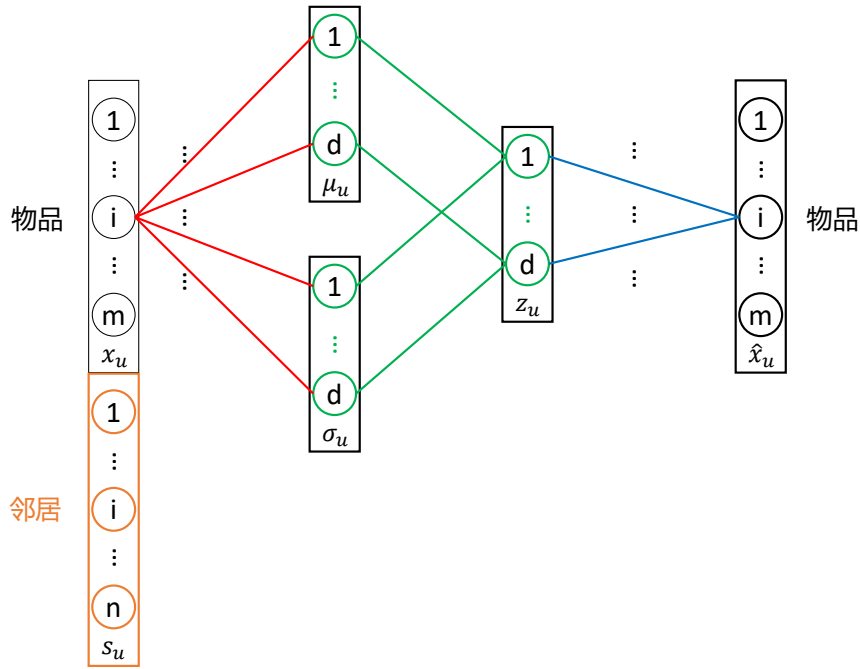


图 3: 输入端拼接邻居多热向量的结构

#### 4.2.4.2 与 $z$ 向量拼接

通过 4.2.3 描述的两种转化邻域信息的方法获得低维的邻居信息向量  $v_u$  后，将  $v_u$  和变分自编码器的  $z_u$  拼接，然后将拼接后的向量输入解码器作最终预测。我们将多热向量接多层感知机获得邻居信息向量并与  $z$  向量拼接的结构命名为 VAEN(H,P)，将通过潜在特征获得邻居信息向量并与  $z$  向量拼接的结构命名为 VAEN(H,E)。这个方式相当于更直接地将邻域信息应用到用户的偏好预测上，结构如图 4 所示。具体表示如下：

$$z'_u = [z_u, v_u] \quad (18)$$

其中， $z'_u \in R^{1 \times (d+d_v)}$  表示拼接后的潜在特征向量。



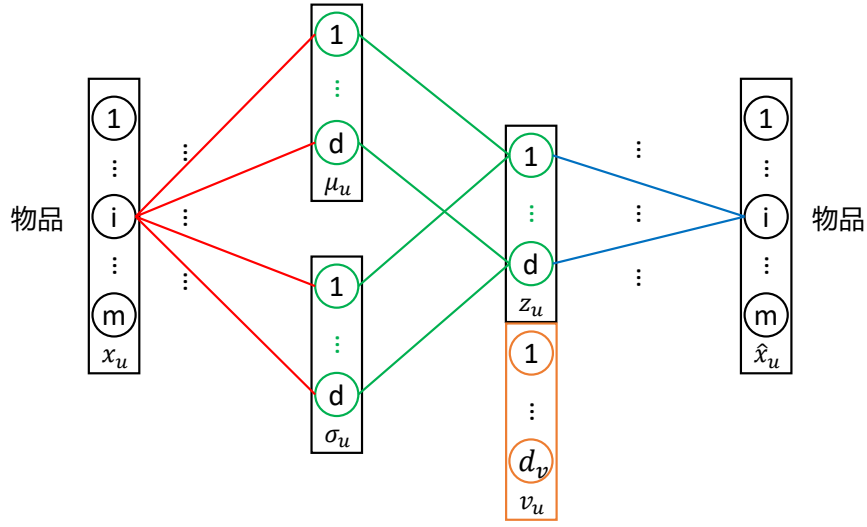


图 4: 和  $z$  向量拼接邻居信息向量的结构

不同于上一个方式接入的是原始邻居数据，这个方式使用的是转化好的邻居信息向量。 $z_u$  在变分自编码器中表示的也是潜在特征向量，所以选择在这里对邻居信息进行接入。拼接后的向量将输入解码器中进行偏好预测，因此邻居信息将在偏好预测的过程中被直接利用。同时，邻居信息向量一定程度上也是当前用户的一个表征，用户的特征能够更好地辅助网络从  $z_u$  中提取用户真正的偏好。

## 5 实验结果分析

### 5.1 数据集

Rec15 是 RecSys 2015 竞赛发布的真实数据集，其记录是和真实的用户购物过程是一致的。原始数据包括了 9,249,729 个会话和 52,739 个物品，其中有 1,150,753 条购买记录和 33,000,944 条点击记录。我们对原始数据进行了如下处理：(i) 对于在一次会话中重复购买或点击了的项目，只保留最早一次交互的记录。(ii) 对于被购买次数少于 5 次的物品和购买记录少于 5 次的会话，我们将其删除。(iii) 对于每个会话，我们将倒数第二个购买记录视为验证集，最后一个购买记录视为测试集，其余记录视为训练集。(iv) 如果验证集或测试集包含训练集中被点击过的物品，我们将删除训练集中的这些物品。处理后的数据集包含了 36,917 个用户，9,621 个物品，训练集包含 159,429 条购买记录、213,332 条点击记录，验证集和测试集分别包括 36,917 条购买记录。处理后的数据集信息如表 1 所示：

表 1: 数据集信息汇总

信息	数量
用户数	36,917
物品数	9621
训练集购买记录	159,429
训练集点击记录	213,332
验证集购买记录	36,917
测试集购买记录	36,917

### 5.2 评价指标和模型参数

为了评估提出的多行为邻域模块在物品推荐任务上的效果，我们采用了四种广泛使用的面向排名的指标，包括正确率、召回率、归一化折损累计增益 (NDCG) 和 1-call<sup>[27]</sup>。我们设定推荐物品的数量的

$N = 5$ ，以上提到的四个指标分别表示为：Prec@5，Rec@5，NDCG@5，1-call@5。

参数设定方面，我们沿用了 VAE++<sup>[20]</sup>中的各项设定，我们固定所有模型的潜在向量维度为  $d = 100$ ，使用的变分自编码器结构只有一层隐藏层，并在隐藏层使用线性激活函数。另外从 0.0001, 0.001, 0.01 中选取学习率，batch size 设为 500，使用 mini-batch Adam 作为优化器，为了防止过拟合，dropout rate 设为 0.5。每训练一个 epoch 即做一次测试，最大训练 epoch 数设为 1000，我们还应用了阈值为 50 的早停机制，即当连续 50 个 epoch 内模型效果都没有变得更好，则终止训练，取最好的一次测试结果。参数范围和数值汇总如表 2 所示。

表 2: 参数范围和数值汇总

参数	范围和数值
d	100
batch size	500
dropout rate	0.5
learning rate	{0.0001,0.001,0.01}
max epoch	1000
early stop patience	50

5.3 实验结果与实验分析

5.3.1 不同的邻域模块结构的效果比较

实验设置：固定邻域构建方式为使用余弦相似度分别计算购买和点击两种行为的用户相似度，然后按  $\alpha = 0.5$  相加，即两种行为权重相同；邻居数设定为 20；邻居潜在信息向量维度  $d_v$  设定为 100。

对多热向量输入端拼接（VAEN(I)）、多热向量接多层感知机拼接  $z$ （VAEN(H,P)）和潜在特征拼接  $z$ （VAEN(H,E)）三种结构进行了实验测试效果，并测试了原始的变分自编码器（Multi-VAE）作为基线来验证邻域模块的加入对效果的提升。实验结果如表 3 所示，其中变分自编码器的结果取自 VAE++<sup>[20]</sup>的论文。

表 3: 不同的邻域模块结构的效果比较

邻域模块结构	Prec@5	Rec@5	NDCG@5	1-call@5
Multi-VAE	0.0511	0.2553	0.1671	0.2553
VAEN(I)	0.0525	0.2624	0.1738	0.2624
VAEN(H,P)	0.0518	0.2592	0.1721	0.2592
VAEN(H,E)	0.0539	0.2695	0.1773	0.2695

从表 3 中的实验结果可以看出三种邻域模块的加入，相比原始的变分自编码器，对结果均有一定的效果提升，其中潜在特征拼接到  $z$  向量上的结构提升最为显著。直接在输入端拼接多热向量的方式没有预先对邻居数据进行转化，而是直接和反馈数据融合到一起通过原始的变分自编码器结构学习；虽然多热向量接多层感知机拼接  $z$  向量的方式对邻居数据经过了多层感知机的转化，但是由于多层感知机的学习能力不够强，最终效果还不如前者。

### 5.3.2 不同的邻域构建方式对效果的影响

实验设置：使用潜在特征拼接  $z$  向量的结构，即 VAEN(H,E) 模型；邻居数设定为 20；邻居潜在信息向量维度  $d_v$  设定为 100。

采用不同的邻域构建方式，测试其对最终结果的影响。对于余弦相似度（CS）和杰卡德指数（JI）两种相似度计算方法，分别测试两种邻域构建方式：1）分别计算购买和点击两种行为的用户相似度，然后按  $\alpha = 0.5$  相加；2）对两种行为的物品集取并集。实验结果如表 4 所示。

表 4: 不同的邻域构建方式对效果的影响

相似度计算方式	Prec@5	Rec@5	NDCG@5	1-call@5
VAEN(H,E) with CS, $\alpha=0.5$	0.0539	0.2695	0.1773	0.2695
VAEN(H,E) with JI, $\alpha=0.5$	0.0538	0.2697	0.1772	0.2697
VAEN(H,E) with CS, $I_u^P \cup I_u^c$	0.0531	0.2651	0.1697	0.2651
VAEN(H,E) with JI, $I_u^P \cup I_u^c$	0.0529	0.2643	0.1693	0.2643

从表 4 中的实验结果可以看出在 Rec15 数据集上分行为单独划分数据集的构建方式明显优于取并集的方式，说明保留各种行为之间的差异性有利于挖掘更深层的邻域信息。在单独划分物品集的情况下，两种相似度计算方法结果相近，影响不大；在取并集的情况下，余弦相似度略优于杰卡德指数。

### 5.3.3 分行为划分物品集时 $\alpha$ 的取值对效果的影响

实验设置：使用潜在特征拼接  $z$  向量的结构，即 VAEN(H,E) 模型；邻居数设定为 20；邻居潜在信息向量维度  $d_v$  设定为 100；相似度计算方法采用余弦相似度。

使用余弦相似度分别计算两种行为的用户间相似度后，按不同的  $\alpha$  取值加权相加，测试其对最终结果的影响。实验结果如表 5 所示。

表 5: 分行为划分物品集时  $\alpha$  的取值对效果的影响

$\alpha$	Prec@5	Rec@5	NDCG@5	1-call@5
1	0.0535	0.2677	0.1755	0.2677
0.8	0.0539	0.2693	0.1768	0.2693
0.6	0.0538	0.2689	0.1767	0.2689
0.5	0.0539	0.2695	0.1773	0.2695
0.4	0.0539	0.2694	0.1776	0.2694
0.2	0.0539	0.2693	0.1776	0.2693
0	0.052	0.2599	0.1709	0.2599

从表 5 中的实验结果可以看出在 Rec15 数据集上，仅使用点击数据构建邻域时效果最差（ $\alpha = 0.0$ ），其次是单独使用购买数据时（ $\alpha = 1.0$ ），而两种反馈数据均使用时，模型的效果会更佳，说明由多种行为构建的邻域会比单一行为的更加精准和有效，达到了最初的预期。权重增大表示该数据在建模时占据更大的重要性，对于点击数据的加入，随着其权重的增大，效果总体上先是增强（ $\alpha = 1.0 \sim 0.5$ ）然后趋于平稳（ $\alpha = 0.5 \sim 0.2$ ），说明了点击数据的有效性。

### 5.3.4 邻居潜在信息向量维度 $d_v$ 对效果的影响

实验设置：使用潜在特征拼接  $z$  向量的结构，即 VAEN(H,E) 模型；固定邻域构建方式为使用余弦相似度分别计算购买和点击两种行为的用户相似度，然后按  $\alpha = 0.5$  相加，即两种行为权重相同；邻居数设定为 20。

使用不同的邻居潜在信息向量维度  $d_v$ ，测试其对最终结果的影响。实验结果如表 6 所示。

表 6: 邻居潜在信息向量维度  $d_v$  对效果的影响

$d_v$	Prec@5	Rec@5	NDCG@5	1-call@5
20	0.0523	0.2613	0.1713	0.2613
40	0.053	0.2648	0.1743	0.2648
60	0.0535	0.2672	0.1757	0.2672
80	0.0537	0.2684	0.1763	0.2684
100	0.0539	0.2695	0.1773	0.2695
120	0.054	0.2698	0.1774	0.2698
140	0.054	0.2702	0.1778	0.2702
160	0.0541	0.2705	0.1779	0.2705
180	0.0541	0.2703	0.178	0.2703
200	0.0542	0.2709	0.1785	0.2709

从表 6 中的实验结果可以看出总体上随着  $d_v$  的增大，效果也在变好。在 20 ~ 100 之间效果增强的趋势比较明显，在 100 后效果增强趋势不明显。说明更大的维度有利于更好地表征用户的潜在特征以及邻居信息。

### 5.3.5 邻居数对效果的影响

实验设置：使用潜在特征拼接  $z$  向量的结构，即 VAEN(H,E) 模型；邻居潜在信息向量维度  $d_v$  设定为 100；固定使用余弦相似度作为相似度计算方法，设定间隔为 2，从邻居数 2 ~ 20 之间分别测试两种邻域构建方式下邻居数对效果的影响：1) 分别计算购买和点击两种行为的用户相似度，然后按  $\alpha=0.5$  相加；2) 对两种行为的物品集取并集。

#### 5.3.5.1 分行为单独计算相似度并相加

表 7 展示了邻居数 2 ~ 20 间分行为单独计算相似度并相加时邻居数对效果的影响。

表 7: 分行为单独计算相似度并相加时邻居数对效果的影响

邻居数	Prec@5	Rec@5	NDCG@5	1-call@5
2	0.0529	0.2645	0.174	0.2645
4	0.0536	0.268	0.1759	0.268
6	0.0536	0.2678	0.1766	0.2678
8	0.0538	0.2689	0.1766	0.2689
10	0.0539	0.2694	0.1772	0.2694
12	0.0535	0.2674	0.1758	0.2674
14	0.0539	0.2695	0.1771	0.2695
16	0.0538	0.2691	0.1771	0.2691
18	0.054	0.2699	0.1773	0.2699
20	0.0539	0.2695	0.1773	0.2695

从表 7 中的实验结果可以看出在邻居数小于 10 时，模型效果随着邻居数增加而变好。在邻居数达到 10 个时效果接近饱和值，和后面 14 ~ 20 个邻居的结果相近。但是邻居数为 12 时效果出现下滑，猜测是在 Rec15 数据集上的特殊情况。

### 5.3.5.2 两种行为的物品集取并集

表 8 展示了邻居数 2 ~ 20 间两种行为的物品集取并集时邻居数对效果的影响。

从表 8 中的实验结果可以看出在邻居数小于 10 时，模型效果随着邻居数增加而变好。在邻居数达到 10 个时效果接近饱和值，和后面 12 ~ 20 个邻居的结果相近。整体的趋势和上面单独划分物品集的实验结果相似。

表 8: 两种行为的物品集取并集时邻居数对效果的影响

邻居数	Prec@5	Rec@5	NDCG@5	1-call@5
2	0.0518	0.2592	0.1659	0.2592
4	0.0523	0.2613	0.1675	0.2613
6	0.0522	0.2612	0.168	0.2612
8	0.0527	0.2633	0.1692	0.2633
10	0.0528	0.2638	0.1695	0.2638
12	0.0529	0.2644	0.1695	0.2644
14	0.0529	0.2645	0.1697	0.2645
16	0.053	0.265	0.1697	0.265
18	0.0529	0.2643	0.1695	0.2643
20	0.0531	0.2651	0.1697	0.2651

### 5.3.6 和其他基于变分自编码器的模型比较

实验设置：由于本文的实验设置沿用了 VAE++<sup>[20]</sup>论文中的设置，下面除了本文提出的方法，其他三个方法的实验结果也是引用 VAE++<sup>[20]</sup>论文中的实验结果。本文提出方法采用的是经过上面实验的探索后得出的最佳效果的结构和参数设定下运行的实验结果，具体为：邻域构建方式为使用余弦相似度分别计算购买和点击两种行为的用户相似度，然后按  $\alpha = 0.5$  相加；邻居数设定为 20；使用潜在

特征拼接  $z$  向量的结构，即 VAEN(H,E) 模型，邻居潜在信息向量维度  $d_v$  设定为 200。

参与比较的方法有基础的变分自编码器（Multi-VAE），阶段式变分自编码器（SVAE）和 VAE++，还有本文提出的邻域增强的变分自编码器（VAEN(H,E)）。值得一提的是， $d_v = 200$  时，VAEN(H,E) 和 SVAE、VAE++ 在  $d = 100$  下主体模型的参数量一致，因为这两个模型均是接入了两个维度为  $d$  的辅助反馈信息向量。实验结果如表 9 所示。

表 9: 本文方法和其他基于变分自编码器的模型比较

方法	Prec@5	Rec@5	NDCG@5	1-call@5
Multi-VAE	0.0511	0.2553	0.1671	0.2553
SVAE	0.0533	0.2664	0.1769	0.2664
VAE++	0.0558	0.2792	0.1861	0.2792
VAEN(H,E)	0.0542	0.2709	0.1785	0.2709

从表 9 中的实验结果可以看出本文提出的方法对比基础的变分自编码器结构有较大的提升；而对于另外两个基于变分自编码器的多行为推荐模型，这两个模型都是直接对辅助数据进行利用来协助目标反馈的预测，建模的视角比较相近。而本文提出的方法是从构建邻域的视角利用多行为数据。最终本文提出方法的效果优于阶段式变分自编码器，而与 VAE++ 还存在一定差距。说明本文提出的多行为邻域模块一定程度上能够有效利用多种反馈来构建用户间邻域，更好地捕捉到用户间的相关性，帮助主体模型更好地对目标反馈进行偏好预测。

## 6 总结与展望

本文针对多行为推荐问题，提出了基于多类反馈构建的邻域模块以及在变分自编码器中的应用。由于基于邻域的方法在单行为推荐中应用非常广泛，而基于多行为数据构建的邻域理论上可以更加精确和有效。具体而言我们首先通过设定的规则来计算多行为数据下用户间的相似度并选取每个用户的邻居，再进一步通过神经网络技术转化邻域信息，最后接入到主体的变分自编码器中辅助目标反馈的偏好学习。通过实验研究了多种反馈下邻域的构建方式、邻域信息的转化方式和邻域信息模块在变分自编码器中的接入方式对模型效果的影响，实验结果表明本文提出的邻域增强的变分自编码器是一种有效的推荐算法。

未来，我们将会考虑把本文提出的多行为邻域模块接入到已有的多行为推荐模型中，作为一个新的利用多行为数据的视角，接入到传统的直接对辅助信息进行利用的模型中，以增强多行为推荐的效果。另外，我们还将继续深耕多行为邻域这个研究方向，试图找到能更好地挖掘和利用多行为邻域信息的方法。

## 参考文献

[1] XIAO B, BENBASAT I. E-commerce product recommendation agents: Use, characteristics, and impact [J]. MIS quarterly, 2007: 137-209.

[2] MILLER B N, ALBERT I, LAM S K, et al. Movielens unplugged: experiences with an occasionally connected recommender system[C]//Proceedings of the 8th international conference on Intelligent user

interfaces. 2003: 263-266.

- [3] LINDEN G, SMITH B, YORK J. Amazon. com recommendations: Item-to-item collaborative filtering [J]. IEEE Internet computing, 2003, 7(1): 76-80.
- [4] MNIH A, SALAKHUTDINOV R R. Probabilistic matrix factorization[J]. Advances in neural information processing systems, 2007, 20.
- [5] KOREN Y, BELL R, VOLINSKY C. Matrix factorization techniques for recommender systems[J]. Computer, 2009, 42(8): 30-37.
- [6] RENDLE S, FREUDENTHALER C, GANTNER Z, et al. BPR: Bayesian personalized ranking from implicit feedback[J]. arXiv preprint arXiv:1205.2618, 2012.
- [7] GAI L, LEI L. Dual collaborative topic modeling from implicit feedbacks[C] // Proceedings 2014 IEEE International Conference on Security, Pattern Analysis, and Cybernetics (SPAC). 2014: 395-404.
- [8] LI G, OU W. Pairwise probabilistic matrix factorization for implicit feedback collaborative filtering[J]. Neurocomputing, 2016, 204: 17-25.
- [9] HE X, ZHANG H, KAN M Y, et al. Fast matrix factorization for online recommendation with implicit feedback[C] // Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. 2016: 549-558.
- [10] PAN W, ZHONG H, XU C, et al. Adaptive Bayesian personalized ranking for heterogeneous implicit feedbacks[J]. Knowledge-Based Systems, 2015, 73: 173-180.
- [11] CHEN X, LI L, PAN W, et al. A survey on heterogeneous one-class collaborative filtering[J]. ACM Transactions on Information Systems (TOIS), 2020, 38(4): 1-54.
- [12] QIU H, LIU Y, GUO G, et al. BPRH: Bayesian personalized ranking for heterogeneous implicit feedback [J]. Information Sciences, 2018, 453: 80-98.
- [13] DING J, YU G, HE X, et al. Improving Implicit Recommender Systems with View Data.[C] // IJCAI. 2018: 3343-3349.
- [14] PAN W, LIU M, MING Z. Transfer learning for heterogeneous one-class collaborative filtering[J]. IEEE Intelligent Systems, 2016, 31(4): 43-49.
- [15] PAN W, YANG Q, CAI W, et al. Transfer to rank for heterogeneous one-class collaborative filtering[J]. ACM Transactions on Information Systems (TOIS), 2019, 37(1): 1-20.
- [16] 陈宪聪, 潘微科, 明仲. 面向异构单类协同过滤的阶段式变分自编码器[J]. 计算机应用, 2021, 41(12): 3499.
- [17] JIN B, GAO C, HE X, et al. Multi-behavior recommendation with graph convolutional networks[C] // Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2020: 659-668.



- [18] XIA L, XU Y, HUANG C, et al. Graph meta network for multi-behavior recommendation[C]// Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2021: 757-766.
- [19] LIANG D, 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.
- [20] MA W, CHEN X, PAN W, et al. VAE++ Variational AutoEncoder for Heterogeneous One-Class Collaborative Filtering[C]// Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining. 2022: 666-674.
- [21] 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.
- [22] WU Y, DUBOIS C, ZHENG A X, et al. Collaborative denoising auto-encoders for top-n recommender systems[C]// Proceedings of the ninth ACM international conference on web search and data mining. 2016: 153-162.
- [23] AGGARWAL C C, WOLF J L, WU K L, et al. Horting hatches an egg: A new graph-theoretic approach to collaborative filtering[C]// Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining. 1999: 201-212.
- [24] DESHPANDE M, KARYPIS G. Item-based top-n recommendation algorithms[J]. ACM Transactions on Information Systems (TOIS), 2004, 22(1): 143-177.
- [25] KINGMA D P, WELLING M. Auto-encoding variational bayes[J]. arXiv preprint arXiv:1312.6114, 2013.
- [26] REZENDE D J, MOHAMED S, WIERSTRA D. Stochastic backpropagation and approximate inference in deep generative models[C]// International conference on machine learning. 2014: 1278-1286.
- [27] VALCARCE D, BELLOGÍN A, PARAPAR J, et al. On the robustness and discriminative power of information retrieval metrics for top-N recommendation[C]// Proceedings of the 12th ACM conference on recommender systems. 2018: 260-268.