

# 基于难分样本挖掘的对抗自编码器推荐系统

魏东, 孙静宇, 海洋

(太原理工大学 软件学院, 太原 030024)

**摘要:** 针对在推荐系统领域中常用数据集的数据分布不平衡、稀疏性大和用户评分偏好不同等问题, 提出了基于难分样本挖掘的对抗自编码器推荐模型; 考虑到用户偏好差异, 使用均模型对数据集进行特征提取处理, 在保留数据统计学特征的同时, 降低了计算复杂度; 之后, 基于三元组损失算法对经过均模型处理的数据集进行难分样本挖掘; 通过对数据集样本进行正负分类, 提升了训练样本质量; 再将正负样本分类后的数据分别作为对抗自编码器的输入, 从重构和对抗两方面共同对评分预测模型进行训练; 同时, 采用 Adam 优化算法为不同参数单独计算更新梯度; 实验结果表明, 该推荐模型显著提升了推荐性能, 多项指标优于基线模型; 基于难分样本挖掘的推荐自编码器推荐系统具有一定实用价值。

**关键词:** 推荐系统; 难分样本挖掘; 用户偏好; 对抗自编码器

## Hard Example Mining Based Adversarial Autoencoder Recommender

Wei Dong, Sun Jingyu, Hai Yang

(College of Software Engineering, Taiyuan University of Technology, Taiyuan 030024, China)

**Abstract:** Commonly used datasets in the field of recommendation suffer from unbalanced data distribution, sparsity and different user rating preferences. All these problems affect the quality of recommendation. Thus, this paper proposed a recommender model combining hard example mining with adversarial autoencoder. Considering the difference of users' preference, Mean Model based triplet loss algorithm was introduced to classify the dataset into positive and negative samples and thus improve the quality of the training data. The application of Mean Model can both reduce computational complexity and retain the statistical feature of original data. Using classified samples, the rating prediction model was trained from both reconstruction and adversarial aspects. Adam optimization algorithm was used to calculate different update gradients for different parameters. Experimental results show that the recommendation model improves the recommendation accuracy significantly, and several performance indexes are better than baseline models. Hard example mining based adversarial autoencoder recommender system has certain practical value.

**Keywords:** recommendation system; hard example mining; user preference; adversarial autoencoder

## 0 引言

传统的推荐算法大致可分为 3 类: 基于内容推荐、协同过滤推荐和混合推荐<sup>[1]</sup>。其中协同过滤算法因其在处理非结构化数据的优越性, 在时下的推荐系统中得到广泛应用和关注。经过学术界和工业界多年的探索和研究, 推荐算法经历了从传统的矩阵分解等方法到现今的与深度学习技术结合的发展历程。

深度学习旨在模拟、建立人脑进行分析学习活动的神经网络, 其模型是一种深层非线性网络, 可以获取比传统算法更深层次的数据本质特征。近些年, 深度学习在图像处理<sup>[2]</sup>、语音识别<sup>[3]</sup>以及自然语言处理<sup>[4]</sup>等领域都取得了很多成果<sup>[5]</sup>, 成为了人工智能领域的一个研究热点, 同时也为推荐系统的研究带来了新的机遇。基于深度学习的推荐

系统通常将各类用户和项目相关的数据作为输入, 利用深度学习网络进行学习和训练, 最终自动为用户输出个性化的推荐结果<sup>[6]</sup>。

本文提出了一种结合难分样本挖掘 (HEM, hard example mining) 和对抗自编码器<sup>[7]</sup> (AAE, adversarial autoencoder) 的深度推荐模型 (HEM-AAE)。难分样本挖掘采用三元组损失算法对项目进行分类, 以此来解决评分数据分布不平衡和稀疏性问题。将不同类别的项目输入对抗自编码器的训练过程, 可以从两方面优化推荐模型。在此基础上通过训练好的模型预测目标用户的项目评分, 使用 TOP-N 算法选择预测评分最高的项目推荐给用户。

## 1 难分样本挖掘

推荐系统研究中常用数据集数据分布不平衡, 稀疏问题较严重, 影响了推荐系统性能稳定性。虽然评分范围固定, 但是用户评分基于个人主观认知, 用户评分标准、偏好不同加剧了数据样本复杂性。所以用户偏好挖掘尤为重要, 故本文引入均模型 (Mean Model) 对数据集做难分样本挖掘 (Hard Negative Mining) 预处理<sup>[10]</sup>。

### 1.1 均模型

由于推荐系统中的常用数据集稀疏性较高, 对计算机

收稿日期: 2020-04-25; 修回日期: 2020-05-21。

基金项目: 山西省科技厅重点研发计划项目 (201803D31226); 山西省研究生教育创新项目 (2019SY117)。

作者简介: 魏东 (1994-), 男, 山西长治人, 硕士研究生, 主要从事数据挖掘与推荐系统方向的研究。

通讯作者: 孙静宇 (1975-), 男, 山西太原人, 博士, 副教授, 主要从事协同 Web 搜索、推荐系统、生物特征识别等方向的研究。

资源消耗较大,故引入均模型<sup>[16]</sup>。均模型生成过程类似于排序二叉树,可以在保留数据统计学特征的情况下极大缓解数据稀疏性,如图 1 所示。

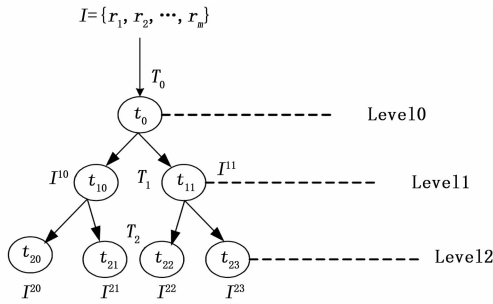


图 1 均模型结构示意图

假设项目评分向量  $I = \{r_1, r_2, \dots, r_m\}$ , 经过变换, 得到此向量的均模型表示为:

$$I^{MM} = \{t_0, (t_{10}, t_{11}), (t_{20}, t_{21}, t_{22}, t_{23}), (t_{30}, t_{31}, \dots), \dots\} \quad (1)$$

其中:  $t_0$  为均模型的根节点,  $(t_{10}, t_{11})$  为均模型第二层的第一和第二个元素,  $t_{lk}$  表示第  $l$  层的第  $k-1$  个结点。结点生成过程如公式 (2):

$$t_{lk} = T_l(I_l) \quad (2)$$

其中:  $T_l(*)$  为第  $l$  层的转换公式 (3):

$$T_l(a_i | a_i \in I_l) = \frac{\sum a_i}{\text{card}(I_l)} \quad (3)$$

当  $k$  为奇数时,  $I_l = \{r_i \in I | r_i > t_{(l-1)g}\}$ ;  $k$  为偶数时,  $I_l = \{r_i \in I | r_i \leq t_{(l-1)g}\}$ ,  $g$  的值为:  $g = \lfloor k/2 \rfloor$ 。在实际应用中, 可以根据需求灵活调整均模型规模, 一般只需三层即可。

### 1.2 三元损失算法

三元组损失算法中, 选任意样本作为基点 Anchor (A), 经过训练, 将同类样本聚集, 称作正样本 Positive (P); 将异类样本疏远, 称作负样本 Negative (N)。三元组损失算法聚焦于确定评分向量正样本和评分向量负样本, 而非评分数值大小, 有助于描述用户评分偏好。在 HEM-AAE 模型训练过程中, 负样本被不断送入模型, 以负样本为反向激励提升模型对正样本的识别、重构能力。使用  $f(x_i^A) \in R^d$  表示用户-评分矩阵中基点项目  $i$  的评分向量, 经过难分样本挖掘后, 使样本特征向量  $f(x_i^P)$  更靠近基点评分向量; 负样本特征向量  $f(x_i^N)$  远离基点评分向量, 如公式 (4):

$$\|f(x_i^A) - f(x_i^P)\|_2 + \alpha < \|f(x_i^A) - f(x_i^N)\|_2 \quad \forall (f(x_i^A), f(x_i^P), f(x_i^N)) \in R^d \quad (4)$$

其中:  $\alpha$  是一个常量, 表示正负样本对训练的边界值。难分样本挖掘代价函数如公式 (5) 所示:

$$L_{HEM} = \sum_i^N [\|f(x_i^A) - f(x_i^P)\|_2 - \|f(x_i^A) - f(x_i^N)\|_2 + \alpha] \quad (5)$$

代价函数采用欧氏距离度量评分向量间距离, 故公式

(5) 恒大于零, 当  $[*]$  大于 0 时, 规定其为损失函数的损失之;  $[*]$  小于 0 时, 规定损失值为 0。

## 2 基于难分样本挖掘的变分自编码器

HEM-AAE 系统框架如图 2 所示。

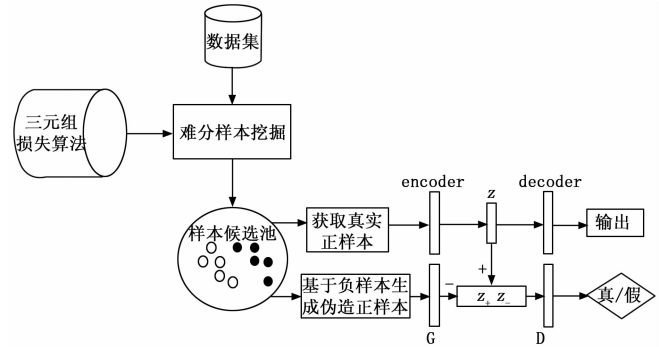


图 2 基于 HEM-AAE 的推荐系统框架

其中 AAE 由自编码器<sup>[8]</sup> (AE, autoencoder) 和生成式对抗网络<sup>[9]</sup> (GAN, generative adversarial networks) 两部分组成。自编码器主要由编码模型 encoder 和解码模型 decoder 构成; 对抗网络由生成模型 G 和判别模型 D 构成。首先, 采用三元组损失算法对数据集进行难分样本挖掘, 经过分类的正、负样本放入样本候选池; 再将正样本和负样本分别作为自编码器 encoder 和对抗网络生成模型 G 的输入, 分别产生正样本隐表示和伪造正样本隐表示; 自编码器的 decoder 根据 encoder 生成的正样本隐表示重构用户评分; 判别模型 D 辨别正样本隐表示和伪造正样本隐表示。

### 2.1 自编码器

自编码器是一种使用误差反向传播算法 (BP, back propagation) 进行训练的前馈神经网络, 结构可简化为如图 3 所示<sup>[11]</sup>。

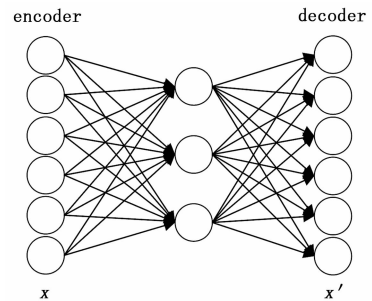


图 3 自编码器结构图

自编码器神经网络由输入层 encoder, 隐藏层和输出层 decoder 构成。通常隐藏层的维度远小于输入层, 输出层的作用是重构输入层, 使用重构误差  $(x, x')$  来表示重构的接近程度。其流程如图 4 所示。

其中 encoder 将输入 INPUT 进行压缩表示, decoder 再将压缩表示进行还原。其数学表达式如式 (5),  $\varphi$  和  $\epsilon$  分别表示 encoder 和 decoder。

$$\varphi, \epsilon = \operatorname{argmin}_{\varphi, \epsilon} L(X, \epsilon(\varphi(X))) \quad (5)$$

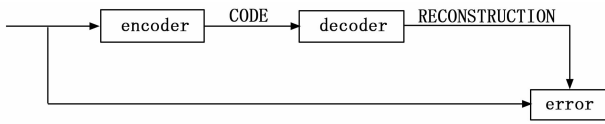


图 4 自编码器神经网络流程图

数据降维和特征提取被认为是自编码器的两个主要实际应用。使用适当的维度和稀疏性约束, 自编码器可以得到比主成分分析或其他类似技术更好的数据投影。

## 2.2 对抗自编码器

如果只通过最小化重构误差来训练模型, 自编码器极有可能学习到一个恒等函数<sup>[1]</sup>, 因此本文引入对抗自编码器进行“对抗”训练。如图 1 所示, 对抗自编码器模型由自编码器和对抗网络两部分组成。训练过程分为也可划分为两阶段: 重构阶段和正则化阶段。本文中, 对抗网络的生成器  $G$  与自编码器的编码模型  $encoder$  使用同一个网络。在重构阶段, 自编码器更新编码模型  $encoder$  和解码模型  $decoder$  以最小化重构误差。在正则化阶段, 判别器  $D$  辨别正样本隐表示  $z_+$  和生成器  $G$  生成的负样本隐表示  $z_-$ , 根据判别结果, 交替更新生成器  $G$  和判别器  $D$ 。对抗自编码器的训练有两个目标: 最小化重构误差和达到对抗网络的相对平衡, 其损失函数如式 (6):

$$L_{AAE} = Reconstruction\ Loss + Adversarial\ Training \quad (6)$$

自编码器的重构输出<sup>[12]</sup>如式 (7) 所示:

$$h_{(r,\theta)} = f(\mathbf{W} \cdot g(\mathbf{V}r + \mu) + b) \quad (7)$$

其中:  $g(\ast)$  使隐藏层的激活函数,  $f(\ast)$  是输出层的激活函,  $\theta = \{\mathbf{W}, \mathbf{V}, \mu, b\}$ , 权重  $\mathbf{W} \in R^{m \times k}$ ,  $\mathbf{V} \in R^{k \times m}$ , 偏置  $\mu \in R^k$ ,  $b \in R^m$ 。输出层对应位置元素被认为是预测值, 即:

$$\hat{r}_{ij} = (h(r^{(i)}; \theta))_u \quad (8)$$

损失函数如公式 (9):

$$L^{RL} = \min_{\theta} \left( \sum_{i=1}^n \|r^{(i)} - h(r^{(i)}; \theta)\|^2 \right) + \frac{\lambda}{2} (\|\mathbf{W}\|_F^2 + \|\mathbf{V}\|_F^2) \quad (9)$$

生成模型  $G$  使用负样本作为输入产生伪造正样本, 对判别器  $D$  进行反向激励, 使得判别器可以更好地识别正样本; 经过优化的判别器同时有利于优化生成器, 生成更好的伪造正样本。对抗网络的训练采用交替优化方法, 即固定  $G$  的参数以更新  $D$  的参数, 然后固定  $D$  的参数去更新  $G$  的参数。Goodfellow 等<sup>[9]</sup>指出, 将生成器  $G$  固定, 可求得唯一的最优判别器:

$$D^* = \frac{p_{data}}{p_g + p_{data}}$$

固定判别器  $D$ , 在  $p_g = p_{data}$  时,  $D^* = 0.5$ , 此时生成器  $G$  达到最优, 即判别器无法区分真实样本和伪造样本<sup>[17]</sup>。损失函数如公式 (10):

$$L^{G^* \cdot D^*} = \min_{\varphi} \max_{\delta} \sum_{i=1}^I (E_{m \sim p_{m^*}(m|u_n)} [\ln D(m|u_n)] +$$

$$E_{d \sim p_{d^*}(d|u_n)} [\ln(1 - D(m|u_n))]) \quad (10)$$

其中:  $m$  为训练集样本,  $U$  表示用户集合,  $u_n$  表示第  $n$  个用户,  $\varphi$  和  $\delta$  分别表示生成器  $G$  和判别器  $D$  的参数。首先更新判别器, 使其最大化正确判别正样本隐向量和伪造正样本隐向量, 如式 (11) 所示:

$$\delta^* = \arg \max_{\delta} \sum_{u \in U} (E_{m_+ \sim p_{m^+}} [\ln D_{\delta}(m_+, m_- | u_n)] + E_{m_- \sim p_{m^-}, m_- \sim G_{\varphi}(m_- | u_n)} [\ln(1 - D_{\delta}(m_g, m_- | u_n))]) \quad (11)$$

其中:  $m_+$ ,  $m_-$  分别代表正样本和负样本,  $m_g$  代表由  $G$  生成的伪造正样本。与判别器  $D$  相反, 生成器  $G$  最小化判别器  $D$  的正确判别概率。故生成模型  $G$  的优化函数如式 (12),  $M$  表示样本集合。

$$\varphi^* = \arg \min_{\varphi} \sum_{m \in M} E_{m \sim G_{\varphi}(m|u_n)} [\ln(1 - D_{\delta}(m_g, m_- | u_n))] \quad (12)$$

在 HEM-AAE 中, 对抗网络和自编码器均使用 Adam 优化算法<sup>[13]</sup>, 即自适应时刻估计方法进行优化训练。Adam 优化算法是一种一阶优化算法。与随机梯度下降法等优化方法最大的区别在于: 通过计算梯度的一阶和二阶矩估计, Adam 算法为每个参数设计了独立的学习率。更新过程如式 (13) ~ (17):

$$m_t = \beta_1 * m_{t-1} + (1 - \mu) * g_t \quad (13)$$

$$n_t = \beta_2 * n_{t-1} + (1 - \nu) * g_t^2 \quad (14)$$

$$\hat{m}_t = \frac{m_t}{1 - \beta_1} \quad (15)$$

$$\hat{n}_t = \frac{n_t}{1 - \beta_2} \quad (16)$$

$$\theta_t = \theta_{t-1} - \frac{\hat{m}_t}{\sqrt{\hat{n}_t} + \omega} * \gamma \quad (17)$$

其中:  $g_t$  是梯度,  $m_t$ 、 $n_t$  分别是对梯度的一阶矩估计和二阶矩估计,  $\hat{m}_t$ 、 $\hat{n}_t$  分别是对  $m_t$  和  $n_t$ ,  $\gamma$  是学习率。更新步长和梯度大小无关, 只和  $\mu$ 、 $\nu$  以及  $\gamma$  有关。学习率会被矩估计动态约束, 使得每个参数都以不同的梯度来更新。而且直接计算梯度的矩估计对内存需求极小, 可应用于大规模深度学习网络。表 1 为本文中 Adam 算法参数初值。

表 1 Adam 算法参数的选取

参数	取值
$\gamma$	0.001
$m_t$	0.9
$n_t$	0.999
$\omega$	$10^{-8}$

## 3 实验结果与分析

### 3.1 实验设计与数据集

HEM-AAE 的  $encoder$ 、 $decoder$ 、生成器  $G$  和判别器  $D$  均采用单层神经网络, 隐藏层神经元个数视不同数据集而不同 (详见 3.2 节), 所有神经网络都使用 Sigmoid 激活函数。运用 Python 编程语言, 通过深度学习框架 TensorFlow 进行神经网络的搭建; 实验中的操作系统为 Ubuntu

18.04 Lts, 在 NVIDIA GTX 1060 6G 显卡上运行。

本文采用 GroupLens 公开数据集 MovieLens-100K、MovieLens-1M<sup>[18]</sup> 来评估 HEM-AAE 的性能。只使用用户 ID, 电影 ID 和评分信息, 其中的每个用户都有 20 个以上的评分记录。两个数据集的统计信息如表 2 所示。实验共进行 5 次, 每次随机选取数据集中的 80% 作为训练集, 20% 作为测试集, 综合 5 次实验结果的平均值得出结论。

表 2 MovieLens 数据集统计信息

	MovieLens-100K	MovieLens-1M
Users	943	6 040
Items	1 682	3 900
Ratings	100 000	1 000 209
Ratings per user	106.04	165.60
Ratings per item	59.45	253.09
Sparsity / %	93.63	95.81

### 3.2 隐藏层规模对算法的影响

本节通过设置不同个数的隐藏层神经单元来研究推荐准确度受隐藏层规模对于 HEM-AAE 模型性能的影响, 神经元个数分别为 [10, 50, 100, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600], 测试结果如图 5 所示。

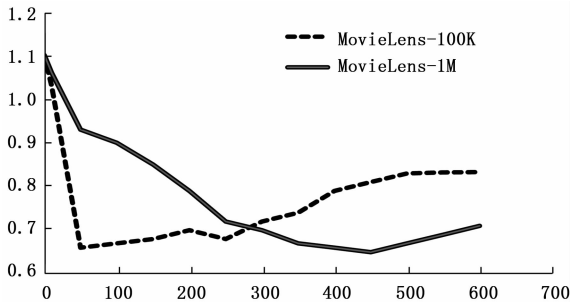


图 5 隐藏层规模对 HEM-AAE 性能的影响

基于测试结果可以发现: 数据、神经网络规模越大, 最佳神经元值也越大。随着神经元个数的增加, 平均绝对误差迅速减小, 但当其超过某个值后, 误差又开始增大。对于不同的数据集, 神经元个数需要经过调试找到最佳值。规模越大的网络过拟合的风险也越大, 所以并不是越大越好。基于测试结果, 分别为数据集 MovieLens-100K、MovieLens-1M 选定的隐层神经元个数为 100 和 400。

### 3.3 性能评价指标

本文预测的是目标用户对待定预测产品的明确的评分, 选用平均绝对误差 (MAE, mean absolute error)、准确率 (precision) 和 NDCG (normalized discounted cumulative gain)。

检测预测准确度, MAE 越小, 说明算法的预测准确度越高。定义如下:

$$MAE = \frac{1}{N} \sum_{i=1}^N |r_i - \hat{r}_i| \quad (18)$$

其中:  $N$  为测试集的大小,  $r_i$  为真实评分,  $\hat{r}_i$  为预测

评分。

准确率表示推荐算法的准确性, 值越高说明推荐的准确性越高, 对于用户  $u$  在生成的推荐的准确率公式为:

$$P_{precision} = \frac{\sum_u |R(u) \cap T(u)|}{\sum_u |R(u)|} \quad (19)$$

其中:  $R(u)$  是训练完毕后为用户  $u$  做出的推荐结果,  $T(u)$  是用户  $u$  在测试集上的真实结果。

NDCG 是一种衡量推荐算法产生的推荐结果的排序质量的评价指标, 该指标考虑到元素之间的相关性, 值越高说明推荐结果的排序质量越好。对于推荐结果中的第  $i$  个结果  $q_i$ , 其 NDCG 值为:

$$NDCG = Z_i \sum_{j=1}^n \frac{2^{y_j} - 1}{\log_2(1 + k_j^i)} \quad (20)$$

其中:  $i$  表示第  $i$  个用户,  $j$  表示推荐列表中第  $j$  个推荐结果,  $k_j^i$  表示为用户  $i$  推荐的结果  $j$  在实际推荐结果中的排名位置,  $y_j$  表示为用户推荐的结果  $j$  在模型生成的推荐列表中的排名位置,  $Z_i$  为标准化因子。

本文选择对比的算法有包括:

1) PMF<sup>[14]</sup>: 概率矩阵分解是将用户物品评价矩阵分解为用户因子和物品因子, 其中假设用户和物品的隐向量服从高斯分布。正则化参数  $\lambda_u$ ,  $\lambda_v$  设置为 0.01 和 0.002 时, PMF 推荐性能最好。

2) PCMM<sup>[15]</sup>, 使用据模型将整体用户集聚类成多个用户子集, 然后在整体上和局部上计算相似度, 利用整合后的相似度预测评分。

3) IRGAN<sup>[16]</sup>, 信息检索生成对抗网络是首个基于生成对抗学习模型的推荐系统。

4) DeepFM<sup>[11]</sup>, 由深度神经网络和因子分解机组成, 可以同时提取到低阶组合特征与高阶组合特征。

### 3.4 结果比较与分析

具体实验结果如表 3、表 4 和表 5 所示, 由表中实验数据可知, HEM-AAE 在各项数据上都相较于 PMF、PCMM 和 IRGAN 都有明显提升。

表 3 各算法/模型在不同数据集上的 MAE

Model	MovieLens-100K	MovieLens-1M
PMF	0.775 2	0.778 9
PCMM	0.752 3	0.751 6
DeepFM	0.732 3	0.721 6
IRGAN	0.725 6	0.721 9
HEM-AAE	0.702 4	0.695 9

如表 3 所示, 在评分预测平均绝对误差方面, 在两个数据集测试中 HEM-AAE 的推荐质量都有很大提高。表 4 实验结果和表 5 实验结果类似, 分别是各算法/模型在 MovieLens-100K 和 MovieLens-1M 数据集上的准确率和 NDCG 指标, 可以看出 HEM-AAE 各项推荐性能指标显著提升, 各算法/模型推荐性能降序序列: HEM-AAE > IRGAN > PCMM > PMF。

表 4 各算法/模型推荐性能比较(MovieLens-100K)

Model	P@3	P@5	P@10	NDGC@3	NDGC@5	NDGC@10
PMF	0.336 9	0.301 3	0.247 7	0.341 9	0.325 7	0.298 4
PCMM	0.345 4	0.312 5	0.271 1	0.341 2	0.326 5	0.301 7
DeepFM	0.393 7	0.373 3	0.340 2	0.404 6	0.394 1	0.368 2
IRGAN	0.407 2	0.371 6	0.321 6	0.419 9	0.399 1	0.372 3
HEM-AAE	0.420 2	0.392 7	0.360 1	0.433 4	0.412 3	0.397 5

表 5 各算法/模型推荐性能比较(MovieLens-100K)

Model	P@3	P@5	P@10	NDGC@3	NDGC@5	NDGC@10
PMF	0.289 5	0.266 4	0.248 3	0.303 7	0.287 8	0.280 4
PCMM	0.300 7	0.294 2	0.281 1	0.322 5	0.301 1	0.293 4
DeepFM	0.373 8	0.361 9	0.331 2	0.402 1	0.390 1	0.374 6
IRGAN	0.380 8	0.368 2	0.342 8	0.418 1	0.406 3	0.394 2
HEM-AAE	0.393 9	0.385 7	0.360 1	0.428 4	0.413 3	0.405 1

### 4 结束语

从实验结果来看, 本文提出的 HEM-AAE 推荐模型有效提高了推荐精度。但是对于新用户没有任何行为记录时, 无法进行推荐, 冷启动问题依然存在。其次由于神经网络是一个黑盒子过程, 无法合理解释在反向传播过程中的具体细节, 所以此算法缺乏一定的可解释性。本文使用的为单层神经网络, 对计算性能要求相对较小, 而在工业界实际操作中, 数据体量远远大于本文实验数据, 所以后期要在分布式集群上进行数据运算, 这样也可以获得更准确的结果。

### 参考文献:

[1] 黄立威, 江碧涛, 吕守业, 等. 基于深度学习的推荐系统研究综述 [J]. 计算机学报, 2018, 41 (7): 191-219.

[2] 李 凯, 彭亦功. 基于生成对抗网络的图像识别改进方法 [J]. 计算机工程与设计, 2019, 40 (2): 199-202, 239.

[3] 蒋畅江, 温登峰, 唐贤伦, 等. 基于改进型轻门控循环单元的语音识别 [J]. 计算机工程与设计, 2019, 40 (11): 3265-3268, 3356.

[4] 朱顺乐. 融合多特征的汉维神经网络机器翻译模型 [J]. 计算机工程与设计, 2019, 40 (5): 292-296, 309.

[5] Le C Y, Bengio Y, Hinton G. Deep learning [J]. Nature, 2015, 521 (7553): 436-444.

[9] 焦媛媛, 田 丰, 石 神, 等. 基于多径的低轨卫星网络路由拥塞控制策略 [J]. 电子设计工程, 2018, 26 (18): 119-123, 128.

[10] 周碧莹, 王爱平, 费长江, 等. 基于强化学习的卫星网络资源调度机制 [J]. 计算机工程与科学, 2019, 41 (12): 2134-2142.

[11] 乔少杰, 韩 楠, 朱新文, 等. 基于卡尔曼滤波的动态轨迹预测算法 [J]. 电子学报, 2018, 46 (2): 418-423.

[6] 张敏军, 华庆一, 贾 伟, 等. 基于深度神经网络的个性化推荐系统研究 [J]. 西南大学学报 (自然科学版), 2019, 41 (11): 104-109.

[7] Makhzani A, Shlens J, Jaitly N, et al. Adversarial Autoencoders [A]. ICLR [C]. 2016; 1482-1490.

[8] Williams S, Hinton G. Learning Representations by Back-propagating Errors [J]. Nature, 1986, 323 (6088): 533-538.

[9] Goodfellow I J, Pouget-Abadie J, Mirza M, et al. Generative Adversarial Nets [A]. NIPS' 2014 [C]. 2014; 2672-2680.

[10] 李广丽, 滑 瑾, 袁 天, 等. 基于用户偏好挖掘生成对抗网络的推荐系统 [J/OL]. 计算机科学与探索: 1-14. doi: 10.3778/j. issn. 1673-9418. 1905090

[11] Guo H F, Tang R M, Ye Y M, et al. DeepFM: a factorization-machine based neural network for CTR prediction [A]. Proceedings of the 26th International Joint Conference on Artificial Intelligence [C]. Melbourne, Australia, 2017: 1725-1731.

[12] Li R, Pan J, Li Z, et al. Single image dehazing via conditional generative adversarial network [A]. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) [C]. IEEE, 2018; 8202-8211.

[13] Sedhain S, Menon A K, Sanner S, et al. Autorec: Autoencoders meet collaborative filtering [A]. International Conference on World Wide Web [C]. ACM, 2015; 111-112.

[14] Kingma D, Ba J. Adam: A method for Stochastic optimization [A]. Proceedings of the 3rd International Conference on Learning Representations [C]. ICLR. 2015; 1-15.

[15] Salakhutdinov R, Mnih A. Probabilistic matrix factorization [J]. Advances in neural information processing systems, 2008; 1257-1264.

[16] 赵瑞皎, 王欣娟, 孙静宇. 社区均模型的 IBCF 算法 [J]. 计算机工程与设计, 2019 (9): 2478-2482

[17] Wang J, Yu L, Zhang W, et al. IRGAN: A Minimax game for unifying generative and discriminative information retrieval models [A]. Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval [C]. SIGIR, 2017; 515-524.

[18] 王万良, 李卓蓉. 生成式对抗网络研究进展 [J]. 通信学报, 2018, 39 (2): 135-148.

[19] Harper F M, Konstan J A. The movie lens datasets [J]. ACM Transactions on Interactive Intelligent Systems, 2016, 5 (4): 1-19.

[12] 吴愉华, 陈立恒, 李世俊. GEO 太阳 X-EUV 成像仪探测器组件热设计 [J]. 计算机仿真, 2019, 36 (9): 263-267.

[13] 徐 垚, 李卓然, 孟金龙, 等. 基于大规模船舶轨迹数据的航道边界提取方法 [J]. 计算机应用, 2019, 39 (1): 105-112.

[14] 王亚平, 彭春艳, 朱名安. 基于 GEO 特发性肺动脉高压芯片数据的生物信息学分析 [J]. 湖北医药学院学报, 2019, 38 (6): 522-528.

[15] 王炳乾, 齐向阳, 周彬斌. 一种改进的 GEO SAR 回波仿真方法 [J]. 雷达科学与技术, 2019, 17 (2): 198-207.

(上接第 116 页)