推荐的对抗性个性化排名*

湘南河
新加坡国立大学xiangnanhe@gmail.com

小玉杜
成都信息工程大学杜克斯me@gmail.com

詹奎鹤
复旦大学
zkhe15@fudan.edu.cn

蔡泰生
新加坡国立大学
dcscts@nus.edu.sg

摘要

项目推荐是个性化的排名任务。为此，许多推荐系统优化具有成对排名目标的模型，例如贝叶斯个性化排名（BPR）。使用矩阵分解（MF） - 推荐中使用最广泛的模型 - 作为演示，我们表明使用BPR对其进行优化会导致推荐模型不稳健。特别是，我们发现结果模型很容易受到模型参数上的对抗性扰动的影响，这意味着概括中可能存在较大的误差。

为了增强推荐器模型的稳健性并从而提高其泛化性能，我们提出了一种新的优化框架，即Adversarial Personalized Ranking（APR）。简而言之，我们的APR通过进行对抗训练来增强成对排名方法BPR。它可以被解释为播放极小极大游戏，其中BPR目标函数的最小化同时防御对手，这增加了对模型参数的对抗性扰动以最大化BPR目标函数。为了说明它是如何工作的，我们通过在用户和项目的嵌入向量上添加对抗性扰动来在MF上实现APR。对三个公共现实世界数据集的广泛实验证明了APR的有效性 - 通过优化MF与APR，它优于BPR，平均相对改善11.2%，并实现项目推荐的最先进性能。我们的实施可在以下网址获得：https://github.com/hexiangnan/adversarial_personalized_ranking。

ccs概念

• 信息系统→推荐系统;信息检索;检索模型和排名;

关键字

个性化排名，成对学习，对抗性训练，矩阵分解，项目推荐

ACM参考格式：
何湘南，何占奎，杜小玉，蔡达生。2018.推荐的对抗性个性化排名。在SIGIR '18：第41届国际ACM SIGIR信息检索研究与开发会议上，2018年7月8日至12日，美国密歇根州安阿伯市。ACM，纽约，纽约，美国，10页。https://doi.org/10.1145/3209978.3209981

*这项工作是在新加坡国立大学的Zhankui He和Xiaoyu Du实习期间完成的。NExT研究由国家研究基金会，总理办公室，新加坡IRC @ SG Funding Initiative支持。

SIGIR'18，2018年7月8日至12日，美国密歇根州安阿伯市
isbn 978-1-4503-5677-2/18/07. $5.00
https://doi.org/10.1145/3209978.3209981

1 介绍

对抗性机器学习的最新进展[30]表明，许多最先进的分类器实际上非常脆弱并且容易受到对抗性的例子的影响，这些例子是通过对来自数据集的输入示例应用小但有意的扰动而形成的。一个典型的例子可以在[15]的图1中找到，它表明通过向熊猫图像添加小的对抗性扰动，训练有素的分类者将图像错误地分类为具有高置信度的长臂猿，而扰动的影响可以很难被人类所感知。这指出了仅在静态标记数据上训练模型的固有限制。为了解决这一局限性并改进模型推广，研究人员随后开发了对抗训练方法，训练模型以正确分类动态生成的对抗性实例[15,25]。

虽然对抗机器学习的鼓舞人心的进步主要集中在可以直观地理解对抗性例子的计算机视觉领域，但到目前为止，还没有关于信息检索（IR）领域中这种对抗性现象的研究。尽管IR中的核心任务是排名，但我们指出许多学习排名（L2R）方法基本上是通过优化分类函数来训练的，例如推荐[28]中的成对L2R方法贝叶斯个性化排名（BPR）等。 [21]。这意味着潜在的IR模型很可能也缺乏稳健性，并且容易受到某些类型的"对抗性示例"的影响。在这项工作中，我们的目标是通过探索项目推荐的对抗性学习方法来填补研究空白，项目推荐是IR中涉及个性化排名的积极和基础研究课题。

然而，直接嫁接从图像域生成对抗性示例的方式是不可行的，因为推荐者模型的输入主要是离散特征（即，用户ID，项目ID和其他分类变量）。显然，将噪声应用于离散特征是没有意义的，这可能会改变它们的语义。为了解决这个问题，我们考虑在更深层次上探索推荐模型的稳健性 - 在其内在层面

模型参数而不是外在输入。使用以BPR训练的矩阵分解（MF）模型[18,20]作为演示（我们将此实例称为MF-BPR），我们研究了其对嵌入参数扰动的鲁棒性。请注意，MF-BPR是一种极具竞争力的项目推荐方法，并且直到最近才被用于许多论文中作为最先进的基线[17]。我们发现MF-BPR不稳健，容易受到参数的对抗扰动的影响。这揭示了BPR培训的弱点，并激励我们开发对抗性学习方法，从而产生更好，更强大的推荐模型。

作为这项工作的主要贡献，我们提出了一种新的对抗性个性化排名（APR）方法来学习推荐模型。以BPR作为构建块，我们引入了一个额外的
APR中的目标函数，用于量化模型在其参数扰动下的损失。APR的表述可以看作是一个极小极大的游戏，其中扰动是优化的
最大化BPR损失，并对模型进行训练
通过对抗性扰动最小化BPR损失和额外损失。具有可区分的推荐模型，整体
APR的框架可以用标准的随机梯度下降进行优化。为了演示它是如何工作的，我们推导了MF的APR求解器，并将该方法称为对抗矩阵分解（AMF）。我们对从Yelp，Pinterest和Gowalla构建的三个公共数据集进行了大量实验，这些数据集代表了各种项目推荐方案。定量和定性分析都证明了对抗性训练对个性化排名的有效性和合理性。具体来说，我们的AMF优于MF-BPR，NDCG和命中率平均显着提高了11%。它也优于最近提出的神经推荐模型[17,35]和IRGAN [31]，并实现项目推荐的最先进性能。

2 预赛

首先描述用于推荐的矩阵分解模型。接下来简要概括了成对学习方法贝叶斯个性化排名。本节的新颖贡献是凭经验证明由BPR（又名MF-BPR）优化的MF模型不稳健，并且易受其参数的对抗性扰动的影响。

2.1 矩阵分解

自几年以来，MF已被公认为推荐的基本但最有效的模型[2,20,40]。作为表示学习的细菌，MF将每个用户和项目表示为嵌入向量。MF的核心思想是估计用户对项目的偏好作为其嵌入向量之间的内在产品。在形式上，让你表示用户，然后我表示一个项目
MF的预测模型表示为：$\hat{y}_{ui}(\Theta) = p_u^T q_i$，其中 $p_u \in R^K$ 和 $q_i \in R^K$ 表示用户的嵌入向量 u 和项目i，K是嵌入向量的大小，也称为嵌入大小。Θ表示MF的模型参数，其由所有用户嵌入和项嵌入向量组成，即，$\Theta = \{p_u, q_i\}$，其中和分别表示所有用户和项的集合。我们使用P和Q来表示嵌入矩阵 $P = \{p_u\}_{u=1}^{U}$，简称 $Q = \{q_i\}_{i=1}^{I}$。

2.2 贝叶斯个性化排名

BPR是成对的L2R方法，并且已被广泛用于优化推荐者模型以实现个性化排名[28]。针对从隐式反馈中学习，它假设观察到的交互应该排名高于未观察到的交互。为此，BPR最大化观察到的交互与未观察到的交互之间的差距。这与逐点方法[2,17]从根本上不同，它们将每个模型预测优化为预定义的基础。形式上，BPR的目标函数（最小化）...是

$$J_{bpr}(d) = \sum_{(u,i,j)\in D} -\ln\sigma(\hat{y}_{ui}(\Theta) - \hat{y}_{uj}(\Theta)) + \lambda_\Theta ||\Theta||^2 \quad (1)$$

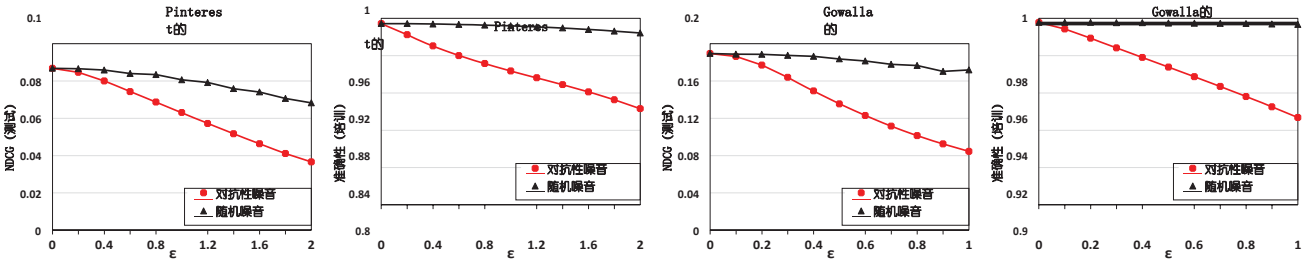
其中σ（·）是S形函数，λΘ是模型特定的正则化参数以防止过度拟合，D表示成对训练实例的集合$D := \{(u, i, j) | i \in I^+ \wedge j \in I \setminus I^+\}$，
其中I^+表示您之前与之交互过的项目集，I_u^+ 和 I 表示整个项目集。自培训人数以来
BPR中的实例非常庞大，通常会优化BPR
通过执行随机梯度下降（SGD）来完成。在获得参数之后，我们可以基于所有项目上的$\hat{y}_{ui}(\Theta)$的值来获得用户u的个性化排序列表。

由于其合理性和易于优化，BPR已经
用于各种场景[6,7,37-39,41]，并在优化推荐模型中发挥重要作用。值得注意的是，BPR的行为可以被解释为分类器 - 给定三元组（u，i，j），它确定（u，i）是否应该具有比（u，j）更高的分数。根据这种解释，（u，i，j）的正实例意味着$\hat{y}_{ui}$应尽可能大于$\hat{y}_{uj}$以获得+1的正确标签;反之亦然，负面实例可以看作标签为0。

2.3 MF-BPR易受对抗性噪音的影响

受到图像分类[15,25,30]中对抗性例子的启发的启发，我们特别感兴趣的是探索BPR是否存在类似现象，因为它也可以被视为一种分类方法用三联（u，i，j）作为输入。与图像域不同，在输入图像中添加小噪声不会改变其视觉内容，BPR的输入是离散ID
功能和更改ID功能将更改输入的语义。例如，如果我们通过破坏用户ID将输入（u，i，j）更改为（u'，i，j），则三元组的语义将完全不同，标签可能会发生变化。因此，为图像分类器生成对抗性示例的现有方法不适合BPR。

由于在输入层添加噪音是不合理的，我们改为
考虑在更深层次上探索BPR的稳健性 - 底层推荐模型的参数。很自然
假设一个健壮的模型应该对小的不敏感
对其参数的扰动;也就是说，只有在强制执行大扰动时，才能显着改变模型行为。为了对所需的扰动进行基准测试，我们使用随机扰动作为基线。如果我们能够找到比随机扰动更有效扰动模型参数的方法，即导致更差的推荐性能，它



(a) 测试NDCG与ε　*(b)* 训练准确性与ε　*(c)* 测试NDCG与ε　*(d)* 训练准确性与ε

图1：在Pinterest和Gowalla上应用对抗性噪声和随机噪声对MF-BPR参数的影响。

意味着模型不那么健壮并且容易受到某些扰动的影响。

设置。考虑到MF在推荐中的主导作用

我们选择MF作为推荐模型，并用BPR进行优化。我们首先训练MF-BPR直到使用SGD收敛。然后，我们比较添加随机扰动和对抗扰动对MF的嵌入的影响。对于对抗性扰动，我们将其定义为旨在最大化BPR目标函数的扰动：

$$\Delta_{进阶} = \max_{\Delta, ||\Delta|| \le \epsilon} l_{bpr}(\hat{d} + \Delta), \tag{2}$$

其中ε控制对抗扰动的大小，表示L2范数，Θ̂是表示当前模型参数的常数集。由于MF是双线性模型而BPR目标函数涉及非线性运算，因此获得关于Δ的精确最大化是难以处理的。受到快速渐变的启发

Goodfellow等人提出的方法。[15]，我们通过将其线性化为Δ来近似目标函数。通过这种近似和最大范数约束，我们可以得到最优Δ：

$$\Delta_{adv} = \epsilon \frac{\Gamma}{||\Gamma||} \quad 其中\Gamma = \frac{\partial l_{bpr}(\hat{d} + \Delta)}{\partial \Delta} \tag{3}$$

由于训练实例的数量很大，我们采样D个未观察到的项目j与观察到的交互（u，i）配对。然后，我们对这组简化的实例，进行实验，以验证对抗性扰动的影响。

结果。图1显示了对我们的ε的不同设置[1]应用对抗和随机扰动对MF-BPR的影响

Pinterest和Gowalla数据集（详见第4.1节）。具体而言，我们展示了NDCG @ 100在保持测试集（图1（a，c））上评估的性能以及减少训练集的分类准确性（图1（b，d））。ε = 0的设置意味着不使用扰动，D表明训练有素的MF-BPR的性能。我们有两个主要观察结果。

- 首先，两个数据集都表明添加对抗性噪音会导致
  比添加随机噪声更显着的性能下降。例如，在Gowalla上，当ε设置为0.4时，应用随机
  扰动使测试NDCG降低1.6％，这对推荐的影响非常小；相反，应用对抗性扰动会使NDCG显着降低21.2％
  – 比随机扰动大13倍。
- 其次，尽管对抗性扰动仅基于部分训练实例D，得出，但它具有显着性
  对推荐表现的不利影响。例如，在Gowalla上，当ε设定为1时，NDCG减少55.4％，而训练精度仅减少5.1％。类似的发现适用于Pinterest数据集，其中测试NDCG的下降和ε= 2时的训练精度分别为57.8％和10.1％。

我们的结果表明，MF-BPR对随机噪声相对稳健，但它很容易受到故意设计的某些扰动的影响。如果推荐者模型是健壮的并且可以很好地预测用户偏好，那么它如何被如此混淆

小规模的扰动？这种有效的对抗性扰动的存在意味着该模型学习了一个复杂的

功能过度拟合训练数据并且不能很好地概括。这促使我们开发用于个性化排名的新训练方法，这可以导致对这种对抗性扰动不敏感的强大的推荐者模型。

3　提出的方法

在本节中，我们首先介绍APR，一种用于个性化排名的对抗性学习框架。然后我们派生出一个通用求解器

适用于基于SGD的APR。最后，我们提出了AMF方法，即使用MF作为推荐器模型的APR实例。

3.1　对抗性个性化排名

我们的目标是设计一个新的目标函数，通过优化它，推荐者模型既适用于个性化排名，又适用于对抗性扰动。由于个性化排名中BPR成对目标的合理性，我们选择它作为构建块。为了增强鲁棒性，即使在出现对抗性扰动（在等式（2）中定义）时，我们也强制执行模型以表现良好。为实现这一目标，我们还优化了模型，以便通过扰动参数最小化BPR目标函数。在形式上，我们将对抗性个性化排名的目标函数定义如下：

$$l_{apr}(d) = l_{bpr}(d) + l_{bpr}(d + \Delta_{进阶}), \tag{4}$$

$$其中\Delta_{进阶} = \arg\max_{\Delta, ||\Delta|| \le \epsilon} l_{bpr}(\hat{d} + \Delta),$$

其中Δ表示模型参数的扰动，ε0控制扰动的大小，Θ̂表示当前模型参数。在这个表述中，对抗性术语

$l_{BPR}(\hat{\Theta} + \Delta_{进阶})$可以看作通过稳定BPR中的分类函数使模型正规化。因此，我们也将其称为对抗正则化器并使用λ来控制其强度。当中间变量Δ使目标函数最大化以使Θ最小化时，APR的训练过程可以表示为

[1]注意，我们对P中的每个嵌入向量强制执行ε的最大范数约束和Q，而不是整个矩阵。

玩极限极小游戏：

$$\Theta^{*},\Delta^{*} = \arg\min_{\Theta}\max_{\Delta,||\Delta||\leq\epsilon} lbpr(d) + lbpr(d$$

其中模型参数Θ的学习算法是最小化播放器，并且获得扰动的过程Δ充当最大化播放器，其旨在识别针对当前模型的最坏情况扰动。两个玩家交替玩游戏直到收敛。由于APR的重点是获得一个好的推荐模型，实际上我们可以通过跟踪模型在验证集上的执行情况来确定何时停止对抗训练。

我们可以看到，与BPR相似，我们对APR的表述导致了一个与模型无关的通用学习框架。只要基础模型yui（Θ）是可微分的，就可以在我们的APR框架下使用反向传播和基于梯度的优化算法来学习。除了BPR中的参数外，还有两个超参数 - ε和λ - 在APR中指定。在下文中，我们提出了基于SGD的APR通用解决方案。

算法1：用于APR的SGD学习算法。

输入：培训数据 Þ对抗性噪声水平ε，对抗正则化器λ，L2正则化器λΘ，学习率η；

输出：模型参数Θ；

1从BPR初始化Θ；

2*虽然不符合停止标准*

3 从D中随机抽取（u，i，j）；

//构建对抗性扰动

4 $\Delta_{进阶}$←等式（8）；

//更新模型参数

5 Θ←等式（11）；

6**结束**

7**返回Θ**

3.2 APR的通用SGD求解器

两种优化策略在推荐中使用最为广泛

dation - 坐标下降（CD）和随机梯度下降（SGD）。CD的典型示例是交替最小二乘[18]，其迭代模型参数并一次更新一个参数。请注意，CD主要用于优化线性模型上的逐点回归损失[2]。当优化目标涉及非线性时，SGD成为默认选择，因为它易于推导更新策略[17,35]。由于APR在其目标函数中涉及非线性函数，并且它具有大量的训练实例（与BPR相同），我们使用SGD优化APR，这比SG更容易实现并且更有效。

SGD的想法是随机绘制训练实例并仅针对单个实例更新模型参数。因此，我们考虑如何针对随机采样的实例（u，i，j）优化模型参数。

步骤1.构建对抗扰动。给出一个

训练实例（u，i，j），构造对抗性扰动$\Delta_{进阶}$的问题可以表示为最大化

$$l_{进阶}((u,i,j)|\Delta) = -\lambda\ln\sigma(yui(\hat{\Theta}+\Delta) - yuj(\hat{\Theta}+\Delta))。 \quad (6)$$

这里$\hat{\Theta}$是表示当前模型参数的常数集。因此，Θ的L2正则化器被丢弃，因为它与Δ无关。然而，对于许多感兴趣的模型，如双线性MF和多层神经网络[17,35]，很难得到$\Delta_{进阶}$的精确最优解。因此，我们采用Goodfellow等人提出的快速梯度法。[15]，对抗训练的共同选择[24,26,34]。该想法是将Δ周围的目标函数近似为线性函数。为了最大化近似线性函数，我们只需要相对于Δ移动目标函数的梯度方向，可以推导为[2]：

$$\frac{\partial l_{进阶}((u,i,j))}{\partial\Delta}\delta = -\lambda(1-\sigma(yui_j(\hat{\Theta}+\Delta)))\frac{\partial\hat{y}_{uij}(\hat{\Theta}+\Delta)}{\partial\Delta}, \quad (7)$$

[2]注意使用的导数规则是：$\frac{\partial\ln x}{\partial x}=\frac{1}{x}$，$\frac{\partial\sigma(x)}{\partial x}=\frac{\sigma(x)}{x}\sigma(x)(1-\sigma(x))$。

其中$yui_j(x) = yui(x) - yuj(x)$。使用max-norm约束$||\Delta||\leq\varepsilon$，我们有$\Delta_{进阶}$的解决方案：

$$\Delta_{adv} = \epsilon\frac{\Gamma}{||\Gamma||} \quad \text{where} \quad \Gamma = \frac{\partial l_{adv}((u,i,j)|\Delta)}{\partial\Delta}. \quad (8)$$

第2步。学习模型参数。我们现在考虑如何学习模型参数Θ。最小化训练实例（u，i，j）的本地目标函数如下：

$$l_{APR}((u,i,j)|\Theta) = -\ln\sigma(yui(\Theta) - yuj(\Theta)) + \lambda\Theta||\Theta||^2 - \lambda\ln\sigma(yui(\Theta+\Delta_{进阶}) yuj(\Theta+\Delta_{进阶}))。 \quad (9)$$

在该问题中，$\Delta_{进阶}$是从等式（8）获得的常数。目标函数相对于Θ的导数如下：

$$\frac{\partial l((u,i,j))}{\partial\Theta} = -(1-\sigma(yui_j(\Theta)))\frac{\partial yuij(\Theta)}{\partial\Theta} + 2\lambda\Theta - \lambda(1-\sigma(yui_j(\Theta+\Delta_{进阶})))\frac{\partial y_{uij}(\Theta+\Delta_{进阶})}{\partial\Theta}. \quad (10)$$

然后我们可以获得Θ的SGD更新规则：

$$\Theta = \Theta - \eta\frac{\partial l_{APR}(((u,i,j)))}{\partial\Theta}, \quad (11)$$

其中η表示学习率。

为了总结APR的SGD求解器，我们在算法1中给出训练过程。在每个训练步骤（第3-5行）中，我们首先随机绘制一个实例（u，i，j）。然后，我们按顺序执行对抗扰动和模型参数的更新规则。

初始化。值得一提的是，模型参数Θ通过优化BPR（第1行）而不是随机初始化来初始化。这是因为当模型参数开始过度拟合数据时，对抗性扰动仅是合理且必要的。当模型不合适时，正常的训练过程足以获得更好的参数。除了预训练与BPR相比，另一种可行的策略是动态调整ε控制训练期间的扰动水平。例如，可以基于保持验证集来学习ε。我们把这个探索留作未来的工作，因为我们发现当前的预测
具有常数ε的训练策略非常有效。

3.3 对抗矩阵分解

为了演示APR的工作原理，我们现在提供一种基于MF的特定推荐解决方案，这是一种基本但非常有效的推荐模型。解决方案简单明了

– 我们首先用BPR培训MF，然后在我们的APR框架下进一步优化它。我们将该方法称为对抗矩阵分解（AMF）。图2说明了我们的AMF方法。由于MF的参数是用户和项目的嵌入向量，我们在嵌入向量上应用对抗扰动。给定（u，i）对，具有扰动的预测模型定义为：

$$\hat{x}_{ui}(\Theta+\Delta) = (\mathbf{q}_i+\Delta_i)^t(\mathbf{q}_i+\Delta_i), \quad (12)$$

其中$\Delta_u \in \mathbb{R}^k$和$\Delta_i \in \mathbb{R}^k$分别表示用户u和项目i的扰动矢量。注意，最大范数约束$\|\Delta\| \leq \varepsilon$是在扰动矢量的水平上强制执行的。要在AMF中应用算法1，我们只需要实现等式（8）（11）。对于等式（8），我们给出了关键导数：

$$\frac{\partial \hat{x}_{uij}(\Theta+\Delta)}{\partial \Delta_u} = \mathbf{q}_i + \Delta_i - \mathbf{q}_j - \Delta_j, \quad \frac{\partial \hat{x}_{uij}(\Theta+\Delta)}{\partial \Delta_i} = \mathbf{p}_u + \Delta_u, \quad \frac{\partial \hat{x}_{uij}(\Theta+\Delta)}{\partial \Delta_j} = -\mathbf{p}_u - \Delta_u. \quad (13)$$

为了实现等式（11），我们给出了如下关键衍生物：

$$\frac{\partial \hat{x}_{uij}(\Theta)}{\partial \mathbf{p}_u} = \mathbf{q}_i - \mathbf{q}_j, \quad \frac{\partial \hat{x}_{uij}(\Theta)}{\partial \mathbf{q}_i} = \mathbf{p}_u, \quad \frac{\partial \hat{x}_{uij}(\Theta)}{\partial \mathbf{q}_j} = -\mathbf{p}_u,$$

$$\frac{\partial \hat{x}_{uij}(\Theta+\Delta_{adv})}{\partial \mathbf{p}_u} = \mathbf{q}_i + \Delta_i - \mathbf{q}_j - \Delta_j,$$

$$\frac{\partial \hat{x}_{uij}(\Theta+\Delta_{adv})}{\partial \mathbf{q}_i} = \mathbf{p}_u + \Delta_u, \quad \frac{\partial \hat{x}_{uij}(\Theta+\Delta_{adv})}{\partial \mathbf{q}_j} = -\mathbf{p}_u - \Delta_u。 \quad (14)$$

3.3.1 AMF的小批量培训。诸如CPU和GPU之类的现代计算单元通常为矩阵式浮点运算提供加速。为了在学习复杂模型中利用这种加速，常见的策略是以小批量方式执行SGD，即，更新一组训练实例上的模型参数而不仅仅是一个实例。实际上，在TensorFlow和Theano等现代工具中实现的许多机器学习方法都应用了小批量优化器。由于AMF玩极限运动游戏并且有两个耦合程序，因此有几种方法可以执行小批量训练。下面我们详细介绍我们如何为AMF进行小批量培训。

首先，给定小批量S，我们随机抽取S训练实例并将小批量称为$\mathcal{D}'$。然后，我们通过最大化小批量的对抗正则化来构建对抗扰动：

$$L_{adv}(\mathcal{D}'|\Delta) = \sum_{(u,i,j)\in\mathcal{D}'} l_{adv}((u,i,j)|\Delta), \quad (15)$$

其中$l_{adv}(u, i, j)|\Delta)$已在等式（6）中定义。对于$\mathcal{D}'$中发生的每个用户和项目[3]，我们通过对$\frac{\partial l_{adv}(d,)}{\partial \Delta}$强制执行max-norm约束来计算其扰动向量。

[3]请注意，该项目包括正项目i和负项目j。有可能的正面1在另一个实例中作为负面项目出现，反之亦然。这个需要考虑到以避免错误。

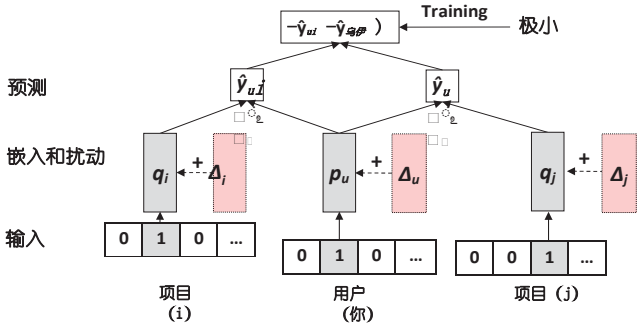


图2：我们的AMF方法的图示。扰动对用户和项目的每个嵌入向量强制执行Δ。

接下来，我们根据小批量更新模型参数$\mathcal{D}'$。小批量的APR目标函数如下：

$$L_{apr}(\mathcal{D}') = \sum_{(u,i,j)\in\mathcal{D}'} l_{apr}((u, i, j), \quad (16)$$

其中$l_{APR}((u, i, j)|\Theta)$已在等式（10）中定义。同样，对于在$\mathcal{D}'$中发生的每个用户和项目，我们执行SGD向上一日期为$\Theta = \Theta - \eta \frac{\partial}{\partial \theta} g$。我们迭代上述两个步骤，直到AMF达到转换状态或验证性能开始退化。

4 实验

由于这项工作的关键贡献是开发一种新的对抗性学习方法APR用于个性化排名，我们的目标是回答以下研究问题通过实验。

RQ1对抗性学习的效果如何？AMF能改善吗？通过进行对抗性学习来超过MF-BPR？

RQ2与最先进的项目相比，AMF的表现如何推荐方法？

RQ3超参数ε和λ如何影响性能以及如何选择最佳值？

接下来，我们首先描述实验设置。然后，我们依次回答上述研究问题来报告结果。

4.1 实验设置

4.1.1 数据集。我们试验了三个公开可用的数据集。表1总结了数据集的统计数据（在所有预处理步骤之后）。这三百万个规模的数据集表示业务，图像和位置登记的不同项目推荐方案。

表1：实验数据集的统计。

数据集	相互作用#	项目#	用户#	稀疏
喊叫	730,790	25,815	25,677	99.89%
Pinterest的	1,500,809	9,916	55,187	99.73%
[illegible]	1,249,703	52,406	54,156	99.96%

1. Yelp的[4]。这是关于企业用户评级的Yelp Challenge数据。我们使用[18]创建的过滤子集来评估项目推荐。我们发现用户可以在不同的时间戳上多次对项目进行评级。自推荐系统

[4]下载地址：https://github.com/hexiangnan/sigir16-eals

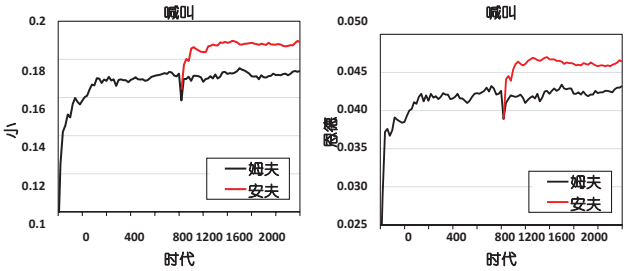


图3：Yelp上MF-BPR和AMF的训练曲线。

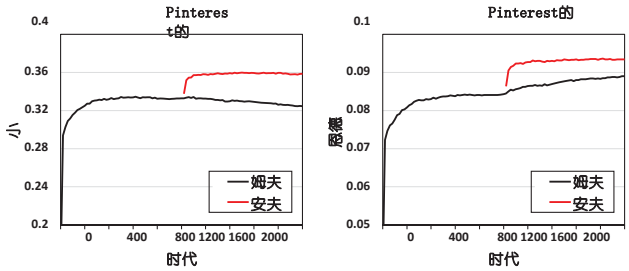


图4：MF-BPR和AMF在Pinterest上的训练曲线。

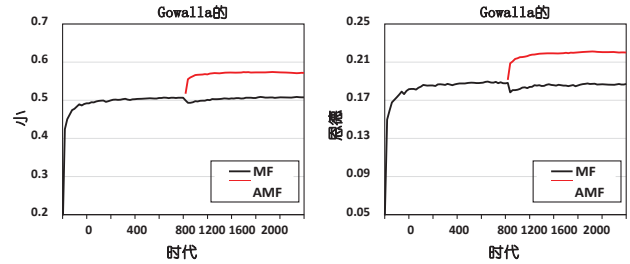


图5：Gowalla上MF-BPR和AMF的训练曲线。

通常旨在推荐用户之前没有消费的项目，我们进一步将重复评级合并到最早的评级。这还可以避免出现在训练集中的测试交互。

2. Pinterest的[5]。这个隐式反馈数据集最初是由[13]构建的基于内容的图像推荐。我们使用[17]创建的过滤子集来评估图像上的协作推荐。由于没有找到重复的交互，我们按原样使用下载的数据集。

3. Gowalla的[6]。这是由[23]构建的登记数据集项目推荐。每次互动都代表用户在Gowalla（一个基于位置的社交网络）的场地上的登记行为。与Yelp的设置相同，我们将重复签到合并到最早的办理登机手续。然后我们过滤掉少于10次互动的项目和少于2次互动的用户。

*4.1.2 评*估协议。我们采用标准的"留一法"协议，该协议已被广泛用于项目推荐评估[2,18,28]。具体来说，对于Yelp和Gowalla中的每个用户，我们将最新的交互作为测试集并在剩余的交互上训练模型。由于Pinterest数据没有时间戳信息，我们随机为每个用户提供交互以形成测试集。

在训练模型之后，我们通过对训练集中用户未交互的所有项目进行排序来为用户生成个性化排名列表。为了研究top-K推荐的表现，我们截断了位置K的排名列表;K的默认设置为100，没有特别提及。然后，我们使用命中率（HR）和标准化折扣累积增益（NDCG）来评估排名列表。HR是基于召回的度量标准，用于衡量测试项目是否位于前K列表中。虽然NDCG对位置敏感，但在较高位置的命中得分较高。对于这两个指标，值越大表示性能越好。我们报告所有用户的平均分数，并执行单样本配对t检验以在必要时判断统计显着性。

*4.1.3 基*线。我们与以下方法进行比较：

- **I**temPop。该方法基于其受欢迎程度对项目进行排名，由训练集中的交互次数证明。这是一种非个性化方法，用于对个性化推荐的性能进行基准测试。
- **M**F-BPR [28]。该方法利用BPR目标函数优化MF。对于项目推荐，这是一种极具竞争力的方法
mendation。我们调整了学习率和L2的系数
正规化建设。
- **C**DAE [35]。此方法扩展了去噪自动编码器以进行项目推荐。已经证明它能够推广几种潜在因子模型。我们使用了作者发布的原始实现[7]，并以与论文中报告的相同方式调整了超参数，包括损失函数，损坏级别，L2正则化和学习率。
- **N**euMF [17]。神经矩阵分解是最先进的
项目推荐方法。它结合了MF和多层感知器（MLP）来学习用户项目交互功能。正如本文所建议的那样，我们用MF预先训练了模型，并调整了隐藏层的深度和L2正则化。
- **I**RGAN [31]。该方法结合了两种类型的模型
对抗性训练，为用户生成项目的生成模型和判断实例是来自真实数据还是生成的判别模型。我们使用了作者发布的实现[8]。我们按照论文的设定
用LambdaFM预先训练发电机[38]。我们分别调整了发生器和鉴别器的学习速率和时期数，我们发现它对其性能有很大影响。进一步调整采样温度并未改善结果，因此我们使用了它们的默认设置。

这组基线代表项目推荐任务的最先进性能。特别是，CDAE和NeuMF是最近提出的神经推荐模型，与MF和FISM等常规浅层方法相比，它们显示出显着的改进[19]。IRGAN利用生成性对抗网络[14]，并在若干IR任务中表现出良好的表现，包括他们的论文中的推荐。

*4.1.4 实*现和参数设置。我们的实现基于TensorFlow，可从以下网址获得：https：//github。COM / hexiangnan / adversarial_personalized_ranking。为了调整超参数，我们从训练交互中随机地为每个用户提供一个交互作为验证集，我们

[5] 下载地址：https：//github.com/hexiangnan/neural_collaborative_filtering

[6] 下载地址：http://dawenl.github.io/data/gowalla_pro.zip

[7] https://github.com/jasonyaw/CDAE(FF554)https://github.com/geek-ai/irgan

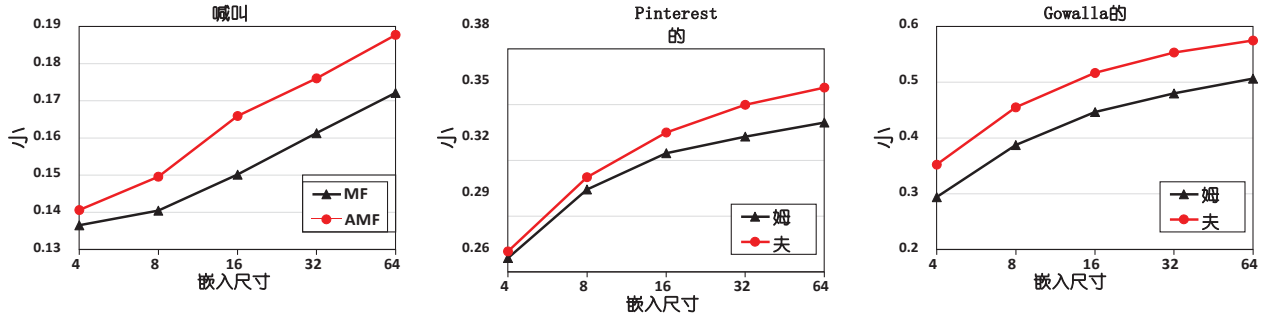


图6：MF-BPR和AMF之间关于不同嵌入尺寸的HR的性能比较。

根据NDCG @ 100选择最优超参数。为了公平比较，所有型号的嵌入尺寸均为64，并使用批量大小为512的小批量Adagrad [12]进行优化;此外，学习率调整为[0.005,0.01,0.05]。对于
AMF，我们调整ε[0.001,0.005,0.01，...，1,5]和λ[0.001,0,01，...，1000] s。32的ize甚至比MF具有更好的嵌入性能使用MF-BPR作为预训练，当所有数据集上的ε= 0.5且λ= 1时，AMF实现了良好的性能。因此，在没有特别提及的情况下，我们会报告AMF在此特定设置下的表现。

4.2 对抗性学习的影响（RQ1）

为了验证对抗性学习的效果，我们首先使用BPR训练MF为1000个时期（主要是收敛），其中每个时期被定义为训练与训练集大小相同的实例数。然后我们继续用APR训练MF，即我们提出的AMF方法;作为比较，我们进一步训练MF与BPR以与APR一致。

1. 培训过程。图3至图5显示了MF的性能和AMF在三个数据集上每20个时期进行评估。我们可以看到所有数据都表现出相同的趋势 - 在1000个时代之后，进一步训练MF与APR导致显着改善，而进一步训练MF与BPR几乎没有改善。例如，在Yelp上（图3），MF-BPR的最佳HR和NDCG分别为0.1721和0.0420，通过APR训练可以提高到0.1881和0.0470。大约10％的相对改善在推荐中非常显着，特别是考虑到潜在的推荐模型保持不变，我们只改变训练方式。

在Gowalla（图5），改进甚至更大 - 分别为HR和NDCG的13.5％和16.8％。在Pinterest（图4），我们注意到MF的HR和NDCG表现出不同的趋势，其中1000个时期HR开始减少而NDCG持续增加。这是可以理解的，因为人力资源和NDCG衡量排名列表的不同方面 - NDCG通过为较高位置的点击分配更高的奖励而HR不是位置敏感的。此外，由于其成对目标，这表明BPR在排名前项中的优势。这一观察结果与[18]在评估top-K推荐方面的发现一致。

2. 改进与模型大小。此外，我们调查对抗性学习的优势是否适用于不同规模的模型。图6显示了MF-BPR和AMF相对于不同嵌入尺寸的性能。请注意，由于空间限制，我们仅显示HR的结果，而NDCG的数字也承认相同的结果。首先，我们可以看到一个明显的趋势，即两种方法的性能随着嵌入量的增加而增加尺寸。这表明，由于增加的建模能力，较大的模型有利于top-K推荐。其次，我们观察到AMF在所有嵌入尺寸的模型上展示了对MF的持续改进。值得注意的是，AMF具有嵌入功能所有数据集的大小为64。这进一步证实了对抗性学习在我们的APR方法中的积极作用。

最后，值得注意的是，与嵌入尺寸较大的设置相比，当嵌入尺寸较小时，AMF的改进不太重要。这意味着当模型很小并且能力有限时，其稳健性不是一个严重的问题。对于容易过度拟合训练数据的大型模型，通过学习对抗性扰动来提高模型的稳健性至关重要，这反过来可以提高其泛化性能。我们相信这种洞察对于推荐任务特别有用，推荐任务通常涉及大量输入特征（例如，用户ID，项目ID以及其他属性和上下文变量）。鉴于如此大的特征空间，即使像分解机器[27]这样的浅嵌入模型也会有大量参数，更不用说神经分解机[16]和深度交叉[29]等更具表现力的深度神经网络。这项工作引入了对抗性学习来解决排名任务，提供了一种新方法来提高大型模型的泛化能力，并有可能改进各种模型。

表2：分别对由BPR和APR训练的MF模型应用对抗性扰动的影响。数字显示NDCG的相对减少。

	$\epsilon = 0.5$		$\epsilon = 1.0$		$\epsilon = 2.0$	
数据集	**BPR**	**四月**	**BPR**	**四月**	**BPR**	**四月**
喊叫	-22.1%	-4.7%	-42.7%	-12.5%	-63.8%	-31.0%
Pinterest的	-9.5%	-2.6%	-25.1%	-7.2%	-55.7%	-23.4%
Gowalla的	-26.3%	-2.9%	-53.0%	-13.2%	-78.0%	-29.2%

3. AMF的稳健性。我们回顾了2.3节中的激励示例，以研究APR训练模型的稳健性。表2显示了对BPR和APR训练的MF模型应用对抗性扰动的影响。

我们可以看到，通过用APR训练MF，与用BPR训练的模型相比，该模型对对抗性扰动变得不那么敏感。例如，在Gowalla上，在0.5到MF-BPR的噪声水平上添加对抗性扰动会使NDCG降低26.3％，而AMF的数量仅为2.9％。这些结果证实了我们的AMF对于对抗性扰动是相当稳健的，这是表明模型具有良好泛化能力的重要特性。

表3：K = 50和K = 100时的Top-K推荐性能。每个设置的最佳结果以粗体突出显示。表明与所有其他方法相比，最佳结果的改善在统计学∗上显着，p <0.01。
最后一列“RI”表示AMF相对于相应基线的平均相对改善。

	Yelp, HR		Yelp, NDCG		Pinterest, HR		Pinterest, NDCG		Gowalla, HR		Gowalla, NDCG		日
	K=50	K=100	K=50	K=100	K=50	K=100	K=50	K=100	K=50	K=100	K=50	K=100	
ItemPop	0.0405	0.0742	0.0114	0.0169	0.0294	0.0485	0.0085	0.0116	0.1183	0.1560	0.0367	0.0428	+416%
姆夫-伯尔	0.1053	0.1721	0.0312	0.0420	0.2226	0.3403	0.0696	0.0886	0.4061	0.5072	0.1714	0.1878	+11.2%
CDAE [35]	0.1041	0.1733	0.0293	0.0405	0.2254	0.3495	0.0672	0.0873	0.4435	0.5483	0.1837	0.2007	+9.5%
伊根［31］	0.1119	0.1765	**0.0361**∗	**0.0465**∗	0.2254	0.3363	0.0724	0.0904	0.4157	0.518	0.1853	0.2019	+5.9%
NeuMF [17]	0.1135	0.1817	0.0335	0.0445	0.2342	0.3526	0.0734	0.0925	0.4558	0.5642	0.1962	0.2138	+2.9%
AMF	**0.1176**∗	**0.1885**∗	0.0350	**0.0465**∗	**0.2375**∗	**0.3595**∗	**0.0741**∗	**0.0938**∗	**0.4693**∗	**0.5763**∗	**0.2039**∗	**0.2212**∗	-

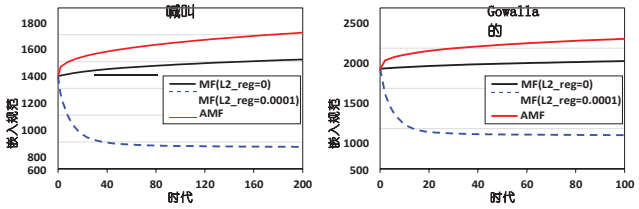


图7：在Yelp和Gowalla的每个训练时期嵌入MF和AMF矩阵的范数。

4. 对抗正规化与L2正则化。APR比BPR改善的原因是因为对抗正则化因素。为了清楚它对参数学习的影响，我们对模型参数进行了一些微观层次的分析。图7显示了在Yelp和Gowalla的每个时期内嵌入MF和AMF的矩阵（即P779）+ Q ^{2}）的范数。作为比较，我们还展示了L2正规化的效果，这是一种推荐用于防止过度拟合的流行技术。

有趣的是，我们发现添加对抗正则化会增加嵌入规范。这是合理的，因为我们将APR中的对抗性扰动限制为具有固定的范数，因此增加嵌入范数有助于减少扰动的影响。然而，简单地通过扩展参数来增加规范是一个简单的解决方案，它不会改善模型的泛化性能。这提供了证据，证明我们提出的学习算法确实以相当有意义的方式更新参数以增强模型的鲁棒性。相反，添加L2正则化会降低嵌入规范以抵抗过度拟合。基于这些，我们得出结论，对抗正则化以与传统的L2正则化不同但更有效的方式改进了模型的推广。

4.3 绩效比较（RQ2）

我们现在将AMF与基线进行比较。表3显示了K设置为50和100时的top-K建议的结果。请注意，我们不会在较小的K处报告结果，因为我们的协议对所有项目进行排序，这使得结果在较小的K处表现出较大的差异。更重要的是，对更大的K进行评估对于从业者来说更有启发性[9]。从表3中，我们有以下主要观察结果：

[9]实际推荐系统通常具有两个阶段[33]，1）候选选择，其选择可能对用户感兴趣的数百个项目，以及2）对候选者重新排序以显示顶部几个结果的排名。第一阶段通常依赖于协同过滤（CF），目的是高召回率。因此，评估具有数百K而不是少量的大K的CF更有启发性。

1. 在大多数情况下，我们的AMF取得了最佳效果。唯一的例外是Yelp，其中IRGAN在NDCG @ 50中的表现优于AMF，在NDCG @ 100中与AMF相当。对于其他情况，AMF在统计学上显着优于其他比较方法，p值小于0.01。这表明AMF实现了最先进的性能
项目推荐。

具体而言，与NeuMF相比 - 最近提出的和非常富有表现力的深度学习模型，AMF的平均改善率为2.9％。这是非常了不起的，因为AMF使用具有更少参数的浅MF模型，这也意味着利用更好的训练算法改进传统浅层方法的潜力。

3. 此外，与IRGAN相比，IRG也在MF上以不同的方式应用对抗性学习，AMF平均上升5.9％。这进一步验证了我们的APR方法的有效性。值得一提的是，APR比IRGAN更有效，更容易训练，需要仔细调整以避免模式崩溃，而APR只需要从BPR初始化。

4. 在基线中，NeuMF表现最佳，它验证了非线性神经网络在学习用户 - 项目交互功能方面的优势。另一种神经推荐模型CDAE表现较弱，仅在Gowalla数据集上显示出相对于MF-BPR的显着改善。在大多数情况下，IRGAN的表现优于MF-BPR，这可归功于其改进的培训流程，因为基础模型也是MF。最后，所有个性化方法都大大优于ItemPop，这表明推荐任务中个性化的必要性。这不是一个新发现，并且已经被许多以前的作品验证过[2,17,28,35,38]。

4.4 超参数研究（RQ3）

我们的APR方法引入了两个额外的超参数ε和λ来分别控制噪声水平和对抗正则化器的强度。在这里，我们展示了两个超参数如何影响性能，并阐明如何设置它们。由于空间限制，我们仅在Pinterest和Gowalla数据集上显示结果，并且Yelp数据集上的结果显示完全相同的趋势。

首先，我们将λ固定为默认值1并改变ε。从图8中可以看出，最佳值约为0.5。当ε太小（例如，小于0.1）时，AMF表现类似于MF-BPR并且仅具有微小的改进。这进一步验证了增加模型对扰动的鲁棒性的积极影响

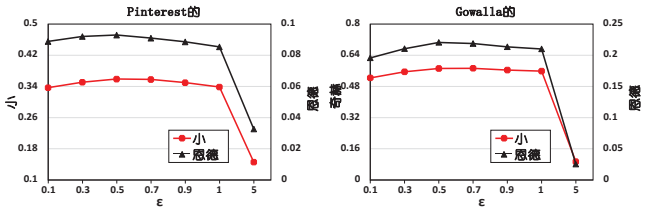


图8：AMF相对于Pinterest和Gowalla上的不同ε值的性能（λ设置为1）。

它的参数。此外，当ε太大（例如，大于1）时，性能急剧下降。这表明过大的扰动会破坏模型参数的学习过程。因此，对于AMF，当我们使用BPR进行预训练时，我们建议的ε设置为0.5。

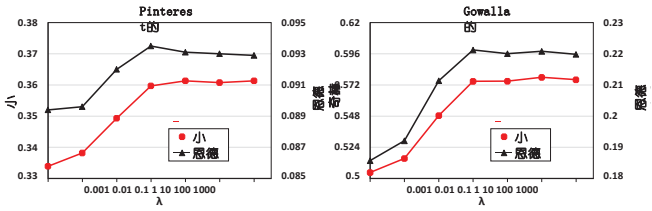


图9：AMF相对于Pinterest和Gowalla上的不同λ值的性能（ε设定为0.5）。

其次，我们将ε固定为0.5并改变λ。图9显示了结果。我们可以看到，当λ小于1时，增加λ会导致逐渐改善。当λ大于1时，进一步增加它既不会改善也不会降低性能直到1000的大值。这意味着当λ足够大以反映对抗效应时，AMF相当不敏感。因此，我们建议将AMF设置为1（或更大的值，例如10）。

5 相关工作

5.1 项目建议

由于可以直接反映用户偏好的这种评级和购买的丰富用户反馈，对项目推荐的研究主要集中于挖掘反馈数据，称为协同过滤（CF）。在各种CF方法中，矩阵分解（MF）[18]是一种特殊类型的潜在因子模型，是一种基本但最有效的推荐模型。通过Netflix挑战推广，CF的早期作品主要集中在显性评级[20,27]。这些工作将推荐任务制定为回归问题，以预测评分。后来，一些研究发现，在评级预测中一个好的CF模型可能不一定在top-K推荐中表现良好[10]，并呼吁推荐研究更多地关注排名任务。

另一方面，对CF的研究逐渐从显性评级转向一级隐式反馈[18,28]。Rendle等人。[28]首先认为项目推荐是一种个性化的排名任务，因此，优化应该针对排名而不是回归进行定制。然后，他们提出了成对学习方法BPR，该方法基于用户对项目对的相对偏好来优化模型。后来，BPR有了被用来优化各种模型[6,7,35,37-39,41]，是推荐中的主导技术。最近，丁等人。[11]通过另外利用电子商务中的观察数据，改进了具有更好的负采样器的BPR。我们提出的APR通过对抗性培训直接增强BPR，有可能改进所有基于BPR的现有推荐系统。

从模型的角度来看，最近有许多努力为CF [1,3,7-9,17,22,32,35,36,41]开发非线性神经网络模型，以利用深度学习。尤其是He等人。[17]论证了MF中固定交互函数（即内积）的局限性，并提出了一种从数据中学习交互函数的神经协同过滤（NCF）框架。然后，他们设计了一个名为NeuMF的NCF模型，该模型统一了MF和MLP在学习交互功能方面的优势。后来，NCF框架被扩展为将用户和项目的邻域[1]和属性[32]合并到POI的模型上下文中。

建议[36]，为多媒体推荐[7]建模图像/视频内容特征，在文本评论[9]中建模方面，为一组用户推荐项目[5]，等等。除了前馈NCF框架之外，还开发了递归神经网络来处理会话感知推荐中的时间信号[3,22]。

5.2 对抗性学习

这项工作的灵感来自对抗性机器学习技术的最新发展[15,24-26,30,34]。简而言之，人们发现，正常的监督训练过程使得一个类比较容易受到对抗性例子的影响[30]，这揭示了一般化中不稳定模型的潜在问题。为了解决这个问题，研究人员随后提出了对抗性训练方法，通过动态生成对抗性实例来增强训练过程[15]。学习这些对抗性的例子可以被视为规范培训过程的一种方式。最近，对抗训练的想法已经扩展到学习深度神经网络中隐藏层的自适应丢失[26]。

关于新兴的对抗性学习领域的现有工作主要集中在图像分类领域。关于排名的对抗性学习的研究很少 - 这是IR的核心任务。与我们最相关的工作是IRGAN [31]，它也采用对抗性学习，更确切地说是GAN框架[14]来解决匹配问题。我们的APR方法与IRGAN根本不同，IRGAN旨在统一生成和判别模型的强度。具体地，在IRGAN的成对公式中，生成器近似相关性分布以生成给定查询（用户）的文档（项）对，并且鉴别器试图区分文档对是来自真实数据还是生成。不幸的是，直觉上很难理解为什么IRGAN-pairwise可以改善项目推荐中的个性化排名（实际上，原始论文和他们发布的代码都只有IRGAN-pointwise用于推荐任务）。

值得注意的是，在推荐系统的文献中，鲁棒性的概念通常是指算法能够抵抗轮廓注入攻击的程度，即试图通过插入用户配置文件来操纵推荐的攻击[4]。由于我们考虑，这一系列研究与我们的工作是正交的

通过使其抵抗其参数的对抗性扰动来改进推荐器模型。通过这种方式，我们可以得到一个更强大和稳定的预测功能，反过来改善

它的泛化性能。据我们所知，此前从未在IR领域进行过探索。

6 结论和未来的工作

这项工作为优化推荐模型提供了一种新的学习方法。我们证明了BPR（推荐中的主导成对学习方法）优化的模型易受其参数的对抗性扰动的影响。这意味着在泛化中用BPR优化的模型可能存在缺陷。为了学习更加健全的个性化排名模型，我们建议对BPR进行对抗性训练，即对抗性个性化排名。我们开发了一种基于SGD的APR通用学习算法，并采用该算法优化MF。在我们的评估中，我们进行了广泛的分析，以展示对抗性学习对个性化排名的高度积极影响。

将来，我们计划将APR方法扩展到其他推荐器模型。首先，我们有兴趣探索更通用的基于特征的模型，如神经分解机[16]和Deep Crossing [29]，它们可以支持广泛的推荐方案，例如冷启动，上下文感知，基于会话的推荐等等。其次，我们将在最近开发的神经CF模型（如NeuMF [17]和基于邻居的NCF [1]）上采用APR，以进一步提高项目推荐的性能。这里的挑战是如何在深层隐藏层上正确使用对抗训练，因为这项工作仅针对浅层MF模型的嵌入层。最后，值得一提的是，我们的APR代表了一种通过使用对抗性训练来改善成对学习的通用方法。成对学习并非特定于推荐，它已被广泛应用于许多其他IR任务，例如文本检索，网络搜索，问答，知识图完成等等。我们将致力于将APR的影响扩展到超出推荐范围的这些领域。

致谢。这项工作得到NExT，新加坡国家研究基金会AI新加坡计划，Linksure Network Holding Pte Ltd和亚洲大数据协会（奖项编号：AISG-100E-2018-002）和国家自然科学的支持。中国基金会批准号：61702300。

引用

[1] T. Bai，J。Wen，J。Zhang和WX Zhao。一种基于交互的邻域神经协同过滤模型。在CIKM，第1979-1982页，2017年。

[2] I.拜耳，X。他，B。Kanagal和S. Rendle。用于从隐式反馈中学习的通用坐标下降框架。在WWW，第1341-1350页，2017年。

[3] A. Beutel，P。Covington，S。Jain，C。Xu，J。Li，V。Gatto和EH Chi。潜在交叉：在循环推荐系统中利用上下文。在WSDM中，第46-54页，第2018页。

[4] R. Burke，MP O'Mahony和NJ Hurley。强大的协作建议书，第961-995页。施普林格美国，波士顿，MA，2015年。

[5] D. Cao，X。他，L。Miao，Y。An，C。Yang和R. Hong。细心的小组推荐。在SIGIR，2018年。

[6] D. Cao，L。Nie，X。他，X。Wei，S。Zhu和T.-S.蔡细历。嵌入因子分解模型，用于联合推荐项目和用户生成的列表。在SIGIR，第585-594页，2017年。

[7] J. Chen，H。Zhang，X。他，L。Nie，W。Liu和T.-S.蔡细历。细心的协作过滤：具有项目和组件级别关注的多媒体推荐。在SIGIR，第335-344页，2017年。

[8] X. Chen，Y。Zhang，Q。Ai，H。Xu，J。Yan和Z. Qin。个性化的关键帧建议。在SIGIR，第315-324页，2017年。

[9] Z. Cheng，Y。Ding，X。他，L。Zhu，X。Song和M. Kankanhalli。NCF：用于评级预测的自适应方面关注模型。在IJCAI，2018年。

[10] P. Cremonesi，Y。Koren和R. Turrin。推荐算法在前n个推荐任务中的表现。在RecSys，第39-46页，2010年。

[11] J. Ding，F。Feng，X。He，G。Yu，Y。Li和D. Jin。通过利用视图数据改进的贝叶斯个性化排名采样器。在WWW，第13-14页，第2018页。

[12] J. Duchi，E。Hazan和Y. Singer。用于在线学习和随机优化的自适应子梯度方法。机器学习研究期刊，12（Jul）：2121-2159,2011。

[13] X. Geng，H。Zhang，J。Bian和T. Chua。学习图像和用户功能以在社交网络中进行推荐。在ICCV，第4274-4282页，2015年。

[14] I. Goodfellow，J。Pouget-Abadie，M。Mirza，B。Xu，D。Warde-Farley，S。Ozair，A.Courville和Y. Bengio。生成对抗网。在NIPS，第2672-2680页，2014年。

[15] I. Goodfellow，J。Shlens和C. Szegedy。解释和利用对抗性的例子。在ICLR，2015年。

[16] X.他和T.-S.蔡细历。用于稀疏预测分析的神经因子分解机器。在SIGIR，第355-364页，2017年。

[17] X.他，L。Liao，H。Zhang，L。Nie，X。Hu和T.-S.蔡细历。神经协同过滤。在WWW，第173-182页，2017年。

[18] X.他，H。张，M.-Y。Kan和T.-S.蔡细历。具有隐式反馈的在线推荐的快速矩阵分解。在SIGIR，第549-558页，2016年。

[19] S. Kabbur，X。Ning和G. Karypis。Fism：top-n推荐系统的因子项相似度模型。在KDD，第659-667页，2013年。

[20] Y. Koren。分解符合邻域：多方面的协同过滤模型。在KDD，第426-434页，2008年。

[21] H.李。学习排名信息检索和自然语言处理，第二版。人类语言技术综合讲座。摩根和Claypool出版社，2014年。

[22] J. Li，P。Ren，Z。Chen，Z。Ren，T。Lian和J. Ma。基于神经注意会话的推荐。在CIKM，第1419-1428页，2017年。

[23] D. Liang，L。Charlin，J。McInerney和DM Blei。在推荐中建模用户曝光。在WWW，第951-961页，2016年。

[24] T. Miyato，AM Dai和I. Goodfellow。半监督文本分类的对抗训练方法。在ICLR，2017年。

[25] S. Moosavi-Dezfooli，A。Fawzi，O。Fawzi和P. Frossard。普遍的对抗性扰动。在CVPR，第86-94页，2017年。

[26] S. Park，J.-K。Park，S.-J。Shin和I.-C.月亮。用于监督和半监督学习的对抗性辍学。在AAAI，2018年。

[27] S. Rendle。分解机器。在ICDM，第995-1000页，2010年。

[28] S. Rendle，C。Freudenthaler，Z。Gantner和L. Schmidt-Thieme。Bpr：来自隐式反馈的贝叶斯个性化排名。在UAI，第452-461页，2009年。

[29] Y. Shan，TR Hoens，J。Jiao，H。Wang，D。Yu和J. Mao。深度交叉：没有手工制作的组合功能的Web级建模。在KDD，第255-262页，2016年。

[30] C. Szegedy，W。Zaremba，I。Sutskever，J。Bruna，D。Erhan，I。Goodfellow，and R. Fergus。神经网络的迷人属性。在ICLR，2014年。

[31] J. Wang，L。Yu，W。Zhang，Y。Gong，Y。Xu，B。Wang，P。Zhang和D. Zhang。Irgan：用于统一生成和判别信息检索模型的极小极大游戏。在SIGIR，第515-524页，2017年。

[32] X.Wang，X。他，L。Nie和T.-S.蔡细历。项目丝绸之路：将信息域中的项目推荐给社交用户。在SIGIR，第185-194页，2017年。

[33] Z. Wang，Z。Jiang，Z。Ren，J。Tang和D. Yin。用于区分电子商务中可替代和互补产品的路径约束框架。在WSDM中，第619-627页，第2018页。

[34] Y. Wu，D。Bamman和S. Russell。关系提取的对抗训练。在ACL中，第1778-1783页，2017年。

[35] Y. Wu，C。DuBois，AX Zheng和M. Ester。用于top-n推荐系统的协同去噪自动编码器。在WSDM中，第153-162页，2016年。

[36] C. Yang，L。Bai，C。Zhang，Q。Yuan和J. Han。桥接协同过滤和半监督学习：poi推荐的神经方法。在KDD，第1245-1254页，2017年。

[37] W. Yu，H。Zhang，X。他，X。Chen，L。Xiong和Z. Qin。以美学为基础的服装推荐。在WWW，第649-658页，第2018页。

[38] F. Yuan，G。Guo，JM Jose，L。Chen，H。Yu和W. Zhang。Lambdafm：使用lambda代理，使用分解机器学习最佳排名。在CIKM，第227-236页，2016年。

[39] F. Zhang，NJ Yuan，D。Lian，X。Xie和W.-Y.嘛。用于推荐系统的协作知识库嵌入。在KDD，第353-362页，2016年。

[40] H. Zhang，F。Shen，W。Liu，X。他，H。Luan和T.-S.蔡细历。离散协同过滤。在SIGIR，第325-334页，2016年。

[41] Y. Zhang，Q。Ai，X。Chen和WB Croft。利用异构信息源进行前n推荐的联合代表学习。在CIKM，第1449-1458页，2017年。