

基于对抗型排序学习的混合推荐算法

许侃¹,吴鑫卓¹,林原^{2*},顾茜³,林鸿飞¹,谢张²

(1.大连理工大学 计算机科学与技术学院,辽宁 大连 116024;

2.大连理工大学 公共管理学院,辽宁 大连 116024;

3.大连理工大学 建设工程学院,辽宁 大连 116024)

摘要:推荐系统可以帮助用户过滤海量信息,单一的推荐算法存在一定的缺陷,基于深度学习的混合推荐通过融入辅助信息可以有效缓解传统推荐算法中数据稀疏的问题,往往可以取得更好的推荐效果。在目前大多数的研究中,针对不同的算法都采用了具体的辅助信息,没有一个统一的混合推荐框架。该文提出了一种基于对抗型排序学习的混合推荐算法——MRecGAN(Mixed Recommendation Generative Adversarial Network);利用排序学习的思想将多个基础推荐算法融合,构建统一的辅助数据,挖掘特征之间的深层关系;并利用生成式对抗网络学习排序函数,通过一个判别器与两个生成器之间的协同对抗,提升各自性能,获得推荐序列;最后利用真实电影Movielens数据集结合辅助数据进行测试。实验结果表明,该模型较好综合了各基础模型的优点,NDCG(Normalized Discounted Cumulative Gain)等指标改善显著,MRR(Mean Reciprocal Ranking)相较于CML(Collaborative Metric Learning)提升32.05%。

关键词:对抗网络;辅助信息;数据稀疏;Movielens;NDCG

中图分类号:TP391 **文献标志码:**A **文章编号:**0253-2395(2024)03-0481-13

Hybrid Recommendation Algorithm Based on Adversarial Learning-to-rank

XU Kan¹, WU Xinzhuo¹, LIN Yuan^{2*}, GU Qian³, LIN Hongfei¹, XIE Zhang²

(1.School of Computer Science and Technology, Dalian University of Technology, Dalian 116024, China;

2.School of Public Administration and Policy, Dalian University of Technology, Dalian 116024, China;

3.School of Infrastructure Engineering, Dalian University of Technology, Dalian 116024, China)

Abstract: The recommendation system can help users filter the massive amount of information. Every single recommendation algorithm has some defects. Mixed recommendation based on deep learning can effectively alleviate the problem of sparse data in traditional recommendation algorithms by incorporating auxiliary information and often achieve better results. In most current studies, specific auxiliary information is used for different algorithms, but there is no unified hybrid recommendation framework. This paper proposed a hybrid recommendation algorithm based on adversarial learning-to-rank: MRecGAN. The idea of learning-to-rank was used to integrate multiple basic recommendation algorithms, built unified auxiliary data, and digged the deep relationship among features. It used generative adversarial networks to learn the sorting function, improved the performance of one discriminator and two generators, and obtained the recommendation sequence. Finally, the real Movielens dataset combined with auxiliary data was used for testing. The experimental results show that the model integrates the advantages of the basic models better. NDCG and other indexes improve significantly, MRR improves by 32.05% compared to CML.

收稿日期:2023-12-15;**接受日期:**2024-01-12

基金项目:国家社会科学基金(20BTQ074)

作者简介:许侃(1981-),男,辽宁大连人,博士,高级工程师,研究方向为信息检索。E-mail:xukan@dltu.edu.cn

* **通信作者:**林原(LIN Yuan),E-mail:zhlin@dltu.edu.cn

引文格式:许侃,吴鑫卓,林原,等.基于对抗型排序学习的混合推荐算法[J].山西大学学报(自然科学版),2024,47(3):481-493. DOI:10.13451/j.sxu.ns.2024005

Key words: adversarial networks; auxiliary information; data sparsity; Movielens; NDCG

0 引言

随着互联网的普及与社交网络的发展,互联网用户从信息匮乏的时代步入了信息过载的时代,用户难以从海量的信息中获取自己所需要的内容。推荐系统为用户推荐其感兴趣的内容,可以过滤大量信息,已经广泛应用于各大领域。在推荐系统中,推荐算法发挥着举足轻重的作用,学术界与工业界都在推荐算法方面取得了一定的成果。而单一的推荐算法往往存在着一些缺陷,无法满足更高的推荐质量的需求,于是提出了混合推荐。

混合推荐通过多种推荐算法的融合,扬长避短,往往可以取得更好的推荐效果。混合推荐含义广泛,将多种算法混合的思想与方法也不尽相同。郭艳红等^[1]将内容预测融入协同过滤方法,冷启动问题得以解决;吕成成等^[2]将KNN(K-NearestNeighbor)与SVM(Support Vector Machine)用于协同过滤,数据稀疏的问题可较好被解决。冀振燕等^[3]融合了多源异构数据,考虑了更为全面的个人兴趣,推荐的准确性进一步提升。胡思才等^[4]将深度神经网络与概率矩阵分解结合,模型在均方误差等指标上得到了改善;Zahra等^[5]提出GHR(SGraph-based Hybrid Recommendation System),构建了基于图的特征与自编码器的推荐系统,准确性优于现有许多算法的同时能够较好解决冷启动问题;Karn等^[6]将混合推荐模型与混合情感分析相结合,推荐效果优于同类不含情感分析的模型。

同时,随着深度学习的迅速发展,推荐领域也迎来了新的发展机遇,深度学习方法可在海量数据中学习特征表示,获得用户与项目之间深层次的非线性关系。但是,用户与项目之间最直接、最明显的关系即是用户对已使用过项目的评分数据,数据的稀疏问题限制了深度学习从数据中学习的能力。而基于深度学习的混合推荐算法,通过融合辅助信息,可从辅助数据中学习用户与项目的隐藏结构,数据稀疏与冷启动等问题可以被有效缓解。但是,在目前

的研究中^[7],包括但不限于基于自编码器(Auto Encoder)、多层感知机(Multilayer Perceptron, MLP)、循环神经网络(Recurrent Neural Network, RNN)与卷积神经网络(Convolutional Neural Networks, CNN)的混合推荐,都需要针对具体的算法设计具体的辅助数据,难以构建一个针对所有数据的、统一的混合推荐框架。

本文利用排序学习的思想,将多种推荐算法运用到该混合推荐框架中,把各推荐算法计算得到的用户对项目的预测得分作为辅助信息,通过特征融合的方式构建新的特征矩阵,作为最终的排序学习推荐数据。同时,本文引入了对抗网络的思想,并对对抗网络进行了改进,新引入一个生成器直接从真实数据中采样,辅助原始生成器训练,进而提出了一种基于对抗型排序学习的混合推荐算法(Mixed Recommendation Generative Adversarial Network, MRecGAN),将对抗网络作为排序学习的学习方法,学习得到效果更好的排序函数,从而为用户推荐排列好先后顺序的项目。该算法提出了一种统一的基于深度学习的混合推荐框架,可以较好解决数据稀疏性问题。该算法在公开数据集Movielens上进行多次测试,不断调节参数,将测试结果与主流推荐算法进行比较,实验结果表明,本文提出的算法在一定程度上还能提高推荐的准确性。

1 相关工作

1.1 推荐算法

推荐算法^[8]作为推荐系统的内核,通过挖掘用户和项目(信息、物品、服务等)之间的二元关系,帮助用户从海量的项目中发现其感兴趣的内容,并生成相应的推荐列表。推荐算法主要包括了传统的推荐算法与基于深度学习的推荐算法。其中,传统的推荐算法可大致分为三类:协同过滤推荐、基于内容的推荐以及混合推荐。而基于深度学习的推荐可以提取出用户与项目的潜在特征,学习到用户与项目之间复杂且潜藏的非线性关系,能够解决传统推荐算法中存在的一些问题。

基于自编码器的协同过滤算法^[9](Autoencoders Recommendation, AutoRec),将自编码器与协同过滤方法相结合,通过编码和解码学习到项目或用户的隐藏表示,并利用隐藏表示预测用户喜好,获取用户对项目的预测得分,完成推荐任务;该模型包含三层神经网络,属于浅层神经网络的范畴,表达高维特征的能力有限,同时易学习到恒等函数导致推荐的准确性降低。基于矩阵分解的推荐算法^[10](Matrix Factorization, MF),分解用户-项目评分矩阵,获得用户与项目的隐藏向量,进而得到预测的评分矩阵;该算法准确性较为稳定,一般采用离线训练的方式,但依赖用户的历史行为,容易发生冷启动等问题。基于神经因子化机的推荐算法^[11](Neural Factorization Machines, NFM),利用神经因子化机对稀疏数据进行特征建模,通过增加全连接层实现特征之间的高层次非线性交互,可在更少的隐藏层条件下提高预测的准确性,同时训练参数更少,训练更加容易。基于循环神经网络的推荐算法^[12](Recurrent Neural Network Recommendation, RNN-Rec),对用户之间的相互依赖情况进行建模,通过记忆并分析用户的历史序列行为,预测其当前时刻可能的行为,进而完成推荐;该算法采用加法叠加状态,存在长期依赖的问题。值得注意的是,以上四种推荐算法均可获得每一个用户对每一个项目的具体预测得分。

1.2 基于深度学习的混合推荐

近年来,深度学习在各个领域都得到了很好的应用,如图像识别、语音识别、语言翻译等,但在推荐领域的应用相比于其他领域相对较少^[13]。数据稀疏与冷启动等问题在传统的推荐算法中表现较为严峻,而基于深度学习的混合推荐算法^[7]可以融合包括用户画像、用户评论、项目内容与社会化标注等数据在内的辅助信息,以缓解上述问题。其中,基于自编码器的混合推荐算法将隐因子作为辅助信息融合到模型中,通过拟合用户-项目评分矩阵,结合重构与评分矩阵误差来构建损失函数,采用梯度下降等方法学习到最终的项目与数据的隐藏向量。协同知识库的嵌入推荐,将项目之间的结构化数据与非结构化数据作为辅助信息,融

入到推荐算法中,利用概率矩阵分解与深度学习方法来学习项目的隐藏表示。基于附加栈式降噪自编码器的混合推荐算法,在每个隐藏层都加入辅助信息,结合矩阵因子分解构建损失函数,由于每个隐藏层都可以加入辅助信息,所以该算法可以融入更加精细的辅助信息。基于卷积神经网络的混合推荐算法,将与项目相关的文本数据作为辅助信息,可结合文本中词语之间的相互作用,帮助学习用户与项目的隐藏表示。基于循环神经网络的混合推荐算法,可结合文本中词语之间的序列关系,完成学习项目的隐藏表示。

1.3 生成式对抗网络

Goodfellow等提出的生成式对抗网络^[14](Generative Adversarial Network, GAN)由两个部分组成,分别是生成器(Generator)和判别器(Discriminator)。其中,生成器在真实数据中采样,生成尽可能与真实数据分布接近的虚假数据,以欺骗判别器,使其不能区分是真实数据还是生成数据;判别器通过输入的样本来判断该样本是真实的数据还是生成器生成的虚假数据。生成器与判别器之间进行零和博弈,采用交替训练的方式提升各自的性能。当算法收敛时,就认为生成器训练达到了最优状态,它生成的数据使得判别器难以判断出真假,此时它就学习得到了真实数据的分布,两者之间达到了纳什均衡。近年来,GAN网络在计算机视觉与自然语言处理领域体现出了巨大的潜力,由于该网络蕴含对抗的思想,在推荐系统中也能取得不错的效果。CHEN等^[15]提出了生成对抗推荐(Generative Adversarial Recommendation, GAR)的通用框架,使得推荐的热启动与冷启动互不影响,在冷启动方面具有更强的综合性能;REN等^[16]将对抗学习引入到顺序推荐中,提出了一种多因素生成对抗网络(Multi-Factor Generative Adversarial Network, MFGAN),并验证了模型的有效性与可解释性;LI等^[17]提出了RecGURU(Adversarial Learning of Generalized User Representations for Cross-Domain Recommendation)算法框架,引入对抗学习,用于训练自编码器与领域判别器,在顺序推荐与跨领域推荐领域表现优异。

1.4 排序学习

排序学习^[18](Learning to Rank)最早在信息检索领域被提出来,它采用机器学习算法学习其核心——排序函数,并利用此函数计算文档与查询的相关分数,以该得分为依据返回查询文档。排序学习的核心与难点在于如何构造这样一个排序函数来计算相关性得分,其基本思路如下;给定文档的训练集合 $D = \{d_1, d_2, \dots, d_n\}$, 文档以三元组的形式构建 $\langle q, d, r \rangle$, 其中 q 为查询, d 为文档的特征集合 $\{f_1, f_2, \dots, f_n\}$, r 为查询与文档的相关性;文档的测试集合 T , 同样采用三元组构建 $\langle q, d, ? \rangle$, 只是查询与文档的相关性未知, 需要利用训练得到的排序函数(即训练模型)进行计算。

排序学习方法一般可分点级(Pointwise)、对级(Pairwise)、列表级(Listwise)三类。点级通过每个文档的特征, 获得文档与查询对之间的具体得分, 根据得分高低完成排序; 对级倾向于文档对之间的偏序关系, 获得对于同一个

查询, 两两文档之间的先后关系, 进而获得整体的先后关系, 它更加接近排序问题的本质; 列表级输入所有相关联文档的集合, 考虑了不同文档的序列关系。大量实验表明^[19-21], 将排序学习的思想用于推荐可以有效优化用户的项目排名, 后文将利用对级中 RankNet 与 LambdaRank 的思想与方法。

2 基于对抗型排序学习的混合推荐算法

本文提出了一种基于对抗型排序学习的混合推荐算法, 将排序学习作为推荐算法的底层模型, 本文将其称为 MRecGAN。其中, 排序学习推荐数据集通过混合的多种推荐算法进行构建, 将各个推荐算法的预测评分向量合并为特征矩阵, 作为辅助信息; 排序模型通过 MRecGAN 对抗训练得到, MRecGAN 将生成式对抗网络的思想用于推荐领域, 通过一个判别器与两个生成器之间的协同对抗, 提升各自的性能。其框架图如图 1 所示。

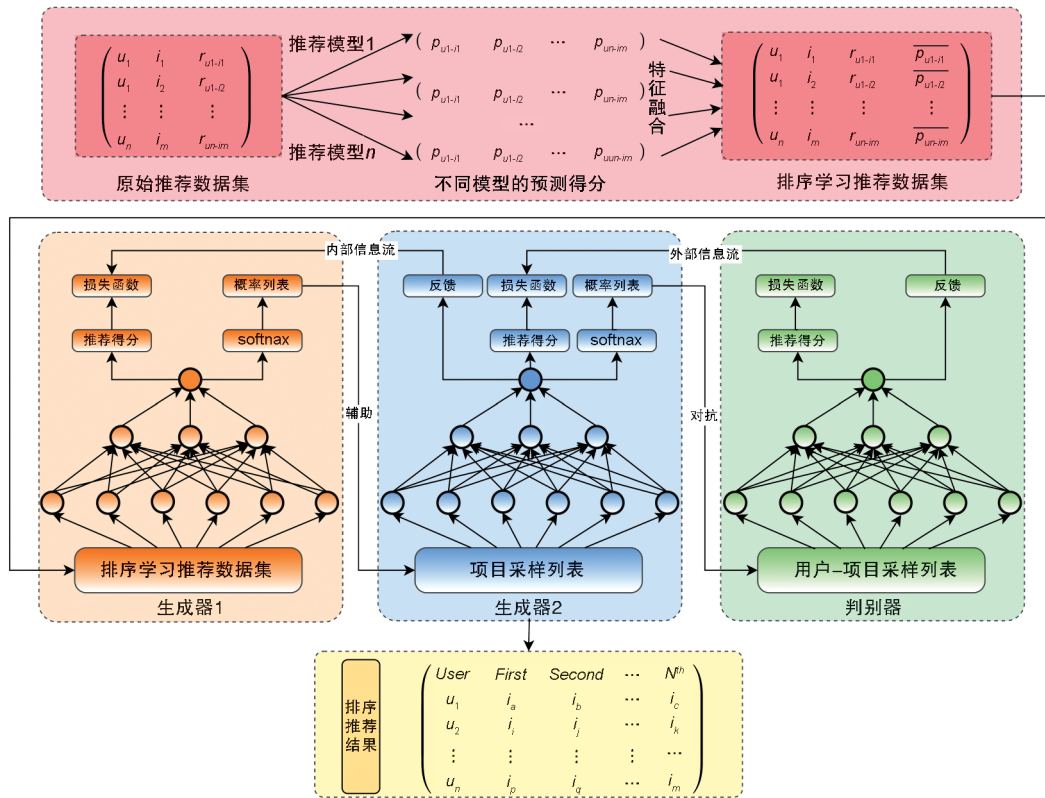


图1 基于对抗型排序学习的混合推荐算法框架

注: 利用现有的推荐模型将原始推荐数据集转化为排序学习推荐数据集, 通过两个生成器与一个判别器协同对抗的方式训练模型, 得到排序推荐结果。

Fig. 1 The structure of the hybrid recommendation algorithm based on adversarial learning-to-rank

2.1 问题转化

本文的底层思路是将排序学习的思想用于解决推荐问题,但排序学习通常用于信息检索领域,故需要首先将推荐问题转化为检索问题。信息检索与推荐是用户获取信息的两种主要方式,检索是主动明确的,推荐是被动模糊的。它们之间虽然存在很多差异,但在技术实现上存在着大量交叠,在逻辑上存在着一一映射的关系。其映射关系如图2所示。

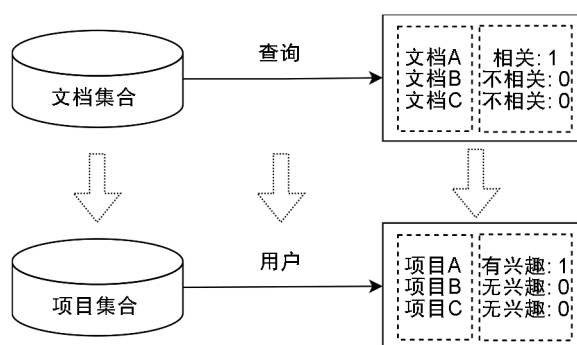


图2 用户-项目与查询-文档的映射关系

注:将排序学习中的“查询-文档-相关性”三元组映射为推荐问题中的“用户-项目-评分”三元组,实现推荐问题向排序问题的转化。

Fig. 2 Mapping relationship between user-item and query-document

基于排序学习的推荐算法^[22]详细思路如下:给定项目集合 $I = \{i_1, i_2, \dots, i_n\}$, 训练得到的排序函数 $f(u, i)$, $u \in U, i \in I$, 其中 U 为用户集合。该排序函数对目标用户 u 产生一个项目排序列表 $\{i_{u1}, i_{u2}, \dots, i_{um}\}$, 并将该列表推荐给用户 u , 其中 i_{ym} 代表向用户 u 推荐的这个项目排在第 m 位 ($m \in [1, n]$)。

通过上述方法,基于排序学习推荐算法的核心问题就转化为了排序模型与排序学习推荐数据集的构建问题。

2.2 构建辅助信息

本文提出了一种统一的基于深度学习的混合推荐框架,通过不同的推荐模型(下文称为基础模型),获取每一个用户对每一个项目的预测评分,并将基础模型获得的预测评分通过特征融合的方式构建为特征矩阵,此矩阵即为本算法所使用的辅助信息,也就是排序学习推荐数据集。通过多个基础模型构建的辅助信

息,可以综合各基础模型的优缺点,增强优点,弥补缺点,往往可以取得更好的推荐效果。值得注意的是,该混合推荐框架是可拓展的,即可以选择更多种类的推荐模型,得到更多维度的预测评分向量,获得更多的隐藏信息。

将推荐数据集输入到基础模型中,可以得到每个模型中用户对物体的预测评分 y_p , 通过多个模型可得到多个不同的预测评分 y_{pn} , 其中 n 代表第 n 个模型。可将这些预测评分以向量 $\gamma = [y_{p1}, y_{p2}, \dots, y_{pn}]$ 的形式进行表示,于是,将推荐数据集中原始三元组 $\langle u, i, y \rangle$ 转换成四元组 $\langle u, i, y_{u-i}, \gamma_{u-i} \rangle$, 其中 $u \in U$, 代表某用户, $i \in I$, 代表某物品, y_{u-i} 为原始数据集中 u 对 i 的真实评分, γ_{u-i} 为 u 对 i 的预测评分向量。

若将每个用户对物品的真实评分与预测评分向量按行排列,同时将用户与物品序号化,即可得到预测评分矩阵 M 与最终的排序学习推荐数据集 N , 其中

$$M = \begin{bmatrix} \gamma_{u1-i1} \\ \gamma_{u1-i2} \\ \dots \\ \gamma_{u2-i1} \\ \dots \\ \gamma_{um-im} \end{bmatrix}, N = \begin{bmatrix} u_1 & i_1 & y_{u1-i1} & \gamma_{u1-i1} \\ u_1 & i_2 & y_{u1-i2} & \gamma_{u1-i2} \\ \dots & \dots & \dots & \dots \\ u_2 & i_1 & y_{u2-i1} & \gamma_{u2-i1} \\ \dots & \dots & \dots & \dots \\ u_n & i_m & y_{un-im} & \gamma_{un-im} \end{bmatrix} \quad (1)$$

2.3 MRecGAN

本文将该推荐算法称为MRecGAN,即融合了多个推荐算法的混合推荐模型,并将GAN网络作为排序学习的学习方法,在原始GAN网络的基础上增加一个新的生成器辅助原始生成器的训练,两个生成器共同与判别器进行对抗,可使模型性能进一步提升。表1列出了本文选取的基础推荐模型的部分特点与缺点,图3展示了推荐结果进行特征融合的方式。

MRecGAN通过多个推荐算法的特征融合,融入辅助信息,得到了排序学习推荐数据集,然后将该数据集输入到对抗网络中。对抗网络包括了两个生成器与一个判别器,它们各自都是一种神经网络,通过神经网络可以拟合得到排序结果,从而计算出损失函数与概率分布列表,同时保留了反馈信息。该对抗网络包含了两种信息反馈,外部信息反馈与内部信息反

表1 混合推荐算法所用基础模型的特点与缺点分析

Table 1 Analysis of characteristics and drawbacks of basic models used in hybrid recommendation algorithms

算法名称	特点	缺点
AutoRec	将自编码器与协同过滤结合	浅层神经网络,特征表达能力有限
MF	矩阵因子分解与协同过滤结合	缺乏用户历史行为时,难以有效推荐
NFM	可学习更高阶的非线性特征交叉	没有考虑H内部元素的相关性
RNN-Rec	可表示用户行为之间的关系	用加法叠加状态,存在长期依赖问题

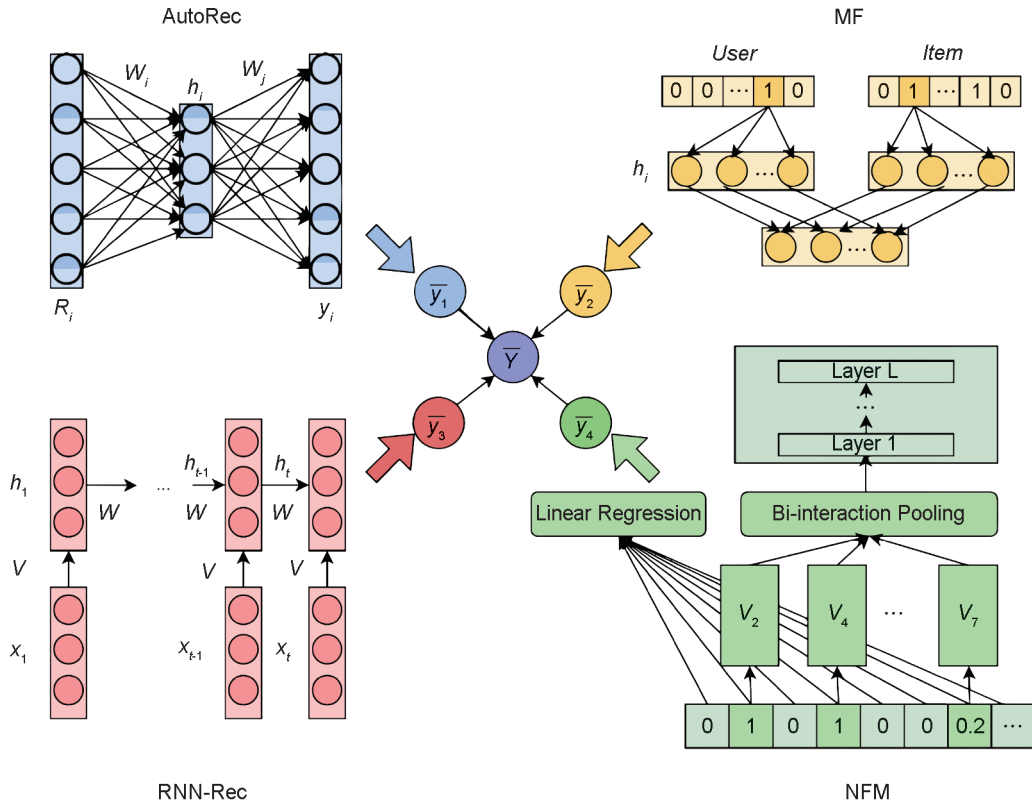


图3 基础模型所得推荐结果的特征融合方式

注:选用AutoRec,MF,RNWN-Rec与NFM四种模型作为基础模型,通过向量拼接的方式融合特征。

Fig. 3 Feature fusion methods for recommended results from basic models

馈,即生成器2与判别器之间的外部信息反馈和生成器1与生成器2之间的内部信息反馈。其中,生成器1与生成器2亦师亦友:生成器2会接受来自生成器1的辅助,即推荐项目的概率列表,用于增强其排序的能力;同时将自身的训练信息,包括梯度下降等内容反馈给生成器1,从而使生成器2能够更好地探索概率分布空间。对于生成器2与判别器,二者亦敌亦友:一方面,生成器2不断产生难以识别是否真实存在的用户-项目对,供判别器进行识别判断;另一方面,判别器将自身训练的反馈信息给到生成器2,指导其训练。综上所述,生成器2与生成器1和判别器均存在两种关系,生成器

1与判别器都是为了让生成器2训练得更好而存在。因此,最终的输出结果也会从生成器2中得到。下面进行详细分析。

其中,生成器1可用 $G_y(i|u, r)$ 表示,生成概率用 $p_y(i|u_n, r)$ 表示。训练过程中,生成器1会不断尝试拟合真实样本分布 $p_{true}(i|u_n, r)$,提升自己的性能,同时为生成器2生成样本的概率列表。

生成器2可用 $G_\theta(i|u, r)$ 表示,生成概率用 $p_\theta(i|u_n, r)$ 表示。生成器2既吸收了来自生成器1的样本概率列表,以生成用于欺骗判别器的假的用户-项目对,又从判别器提供的外部信息反馈中学习了每个用户-项目对的判别难易程度,以不断改进自身性能。

判别器可用 $D_\delta(i, u)$ 表示, 其输入为真实或生成的项目与相应的用户, 输出为用户对项目的匹配程度或者感兴趣程度。判别器不断地从生成器 2 提供的用户-项目对中判断该用户是否真的对该项目感兴趣, 并把其判断的情况反馈给生成器 2。

整体对抗型排序学习的混合推荐算法的目标函数包括两部分:

(1) 内部目标函数: 通过最大化交叉熵损失来改进生成器 2 的性能。训练过程中, 生成器 1 对生成样本进行采样。

$$J^G = \min_\gamma \max_\theta \sum_{n=1}^N \left(E_{i \sim p_{\text{true}}(i|u_n, r)} \log G(i|u_n) + E_{i \sim p_\gamma(i|u_n, r)} \log(1 - G(i|u_n)) \right), \quad (2)$$

其中 i 代表输入到网络中的样本; u_n 代表用户, n 为用户编号; r 代表输入样本的类别, r 为 1 代表真实样本, r 为 0 代表生成样本; $p_{\text{true}}(i|u_n, r)$ 为真实样本 i 的概率分布, $p_\gamma(i|u_n, r)$ 为生成样本 i 的概率分布; γ 与 θ 分别为生成器 1 与生成器 2 的参数。

首先固定参数 θ , 即不对生成器 2 的参数进行更新, 对生成器 1 进行优化。即只更新 $E_{i \sim p_\gamma(i|u_n, r)} \log(1 - G(i|u_n))$, 当 $r=0$ 时, 生成器 1 生成样本 i , 希望生成器 2 的评分函数 $G(i|u_n, r)$ 为其打高分, 使整体期望值最小。然后, 固定参数 γ , 即不对生成器 1 的参数进行更新, 对生成器 2 进行优化。 $E_{i \sim p_{\text{true}}(i|u_n, r)} \log G(i|u_n, r)$ 中, 当 $r=1$ 时, 为真实样本, 生成器 2 希望评分函数 $G(i|u_n, r)$ 为其打高分; $E_{i \sim p_\gamma(i|u_n, r)} \log(1 - G(i|u_n))$ 中, 当 $r=0$ 时, 由生成器 1 筛选样本, 生成器 2 希望评分函数 $G(i|u_n, r)$ 为其打低分, 使整体期望最大。

(2) 外部目标函数: 通过最大化交叉熵损失来改进判别器性能, 希望判别器能够将真实样本判别为 1, 生成样本判别为 0。

$$J^{G, D} = \min_\theta \max_\delta \left(E_{i \sim p_{\text{true}}(i|u_n, r)} \log D(i|u_n) + E_{i \sim p_\gamma(i|u_n, r)} \log(1 - D(i|u_n)) \right), \quad (3)$$

其中 i 代表输入到网络中的样本; u_n 代表用户, n 为用户编号; r 代表输入样本的类别, r 为 1 代表真实样本, r 为 0 代表生成样本; $p_{\text{true}}(i|u_n, r)$ 为真实样本 i 的概率分布, $p_\gamma(i|u_n, r)$ 为生成器 2 生

成样本 i 的概率分布; θ 与 δ 分别为生成器 2 与判别器的参数; $D(i|u_n)$ 代表判别器给“用户 n -样本 i 对”的打分, 其输出结果区间为 $[0, 1]$, 于是 $\log D(i|u_n)$ 与 $\log(1 - D(i|u_n))$ 的取值范围为 $(-\infty, 0]$ 。

对于生成器 2, 我们希望通过训练, 生成让用户感兴趣的非真实存在的项目, 即在 $E_{i \sim p_\gamma(i|u_n, r)} \log(1 - D(i|u_n))$ 中让 $D(i|u_n)$ 给出高分, 使 $\log(1 - D(i|u_n))$ 变小, 所以生成器 2 的训练实质是最小化 $J^{G, D}$ 。对于判别器, 我们希望通过训练, 让其能够判别出哪些是生成器 2 生成的项目, 哪些是真实存在的项目。也就是说, 当判别器接受到生成样本时, 判别器给出低分, 接收到真实数据时, 判别器给出高分, 即最大化 $J^{G, D}$ 。

首先固定参数 θ , 即不对生成器 2 的参数进行更新, 而是对判别器进行优化。当 $r=0$ 时, 为生成样本, 生成器 2 筛选样本 i , 希望生成尽可能真实的样本, 使整体期望值最大; 当 $r=1$ 时, 为真实样本, 判别器希望得高分, 找到当前最优的判别器。为了使公式最大化, $D(i|u_n)$ 输出的值接近 1 时, 代表判别器成功识别了真实数据与生成数据。然后固定参数 δ , 即不对判别器的参数进行更新, 而是对生成器 2 进行优化。此时只讨论生成样本, 不用讨论真实样本, 即 $E_{i \sim p_{\text{true}}(i|u_n, r)} \log D(i|u_n)$ 不会更新, 只需更新 $E_{i \sim p_\gamma(i|u_n, r)} \log(1 - D(i|u_n))$ 。当 $r=0$ 时, 由生成器 2 筛选样本 i , 生成器 2 希望得高分, 使整体期望值最小。生成器 2 与判别器相互对抗, 交替更新, 最终二者都达到最优状态。

2.4 生成器与判别器的训练

2.4.1 训练生成器 1

生成器 1 利用 LambdaRank^[23] 作为基础模型, 将 LambdaRank 的损失函数与生成器 1 的生成概率 $p_\gamma(i|u, r)$ 相结合, 共同作为生成器 1 的目标函数, 将其记作 $g_{\text{yscore}}(i, u)$ 。

令 $J^{G_\gamma}(u_n) = E_{i \sim p_\gamma(i|u_n, r)} \log(1 - G(i|u_n))$, 则

$$\gamma^* = \operatorname{argmax}_\gamma \sum_{n=1}^N \left(E_{i \sim p_\gamma(i|u_n, r)} \right) \times \left[\log(1 - G_\gamma(i|u_n)) \right] = \operatorname{argmax}_\gamma \sum_{n=1}^N J^{G_\gamma}(u_n). \quad (4)$$

对 $J^{G_2}(u_n)$ 求 γ 的偏导, 并近似得到:

$$\Delta_\gamma J^{G_2}(u_n) = \Delta_\gamma E_{i \sim p_\gamma(i|u_n, r)} \log(1 - G_\gamma(i|u_n)) = E_{i \sim p_\gamma(i|u_n, r)} \Delta_\gamma \log p_\gamma(i|u_n, r) \log(1 - G_\theta(i|u_n)) \approx \frac{1}{K} \sum_{i=1}^K \Delta_\gamma \left[\underbrace{\log p_\gamma(i|u_n, r)}_{\text{ListLoss}} \underbrace{\log(1 - G_\theta(i|u_n))}_{\text{Reward}} \right] \quad (5)$$

其中 K 为特定用户的商品编号。通过以上推导, 可得到最优 γ 为:

$$\gamma = N \left(\frac{1}{1 + e^{s_i - s_j}} \right) (2^i - 2^j) \left(\frac{1}{\ln(1+i)} - \frac{1}{\ln(1+j)} \right), \quad (6)$$

其中 N 为最大折损累计增益 (DCG) 的倒数。生成器 1 将 Pointloss 作为其损失函数。

$$\log G_\theta(i|u_n) = \text{Pointloss} = \sum_{n=1}^K \log \sigma(f_\theta(I_n)). \quad (7)$$

2.4.2 训练生成器 2

生成器 2 接收来自生成器 1 的困难样本, 同时为判别器提供假样本用于欺骗判别器, 生成器 2 拟合 $g_{\text{score}}(i, u)$ 函数并生成概率 $p_\theta(i|u_n, r)$ 。值得注意的是, 通常生成器在前期的训练中表现较差, 所以新引入了生成器 1 来辅助生成器 2 的训练。通过信息反馈, 可得到最优 θ^* 。

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \sum_{n=1}^N \left(E_{i \sim p_\theta(i|u_n, r)} \right) \left[\log(1 - D(i|u_n)) \right] \approx \underset{\theta}{\operatorname{argmax}} \frac{1}{K} \sum_{i=1}^K \left[\underbrace{\Delta_\theta \log p_\theta(i|u, r)}_{\text{Pointloss}} \underbrace{\log(1 - D(i|u_n))}_{\text{Reward}} \right]. \quad (8)$$

2.4.3 训练判别器

判别器通过拟合 $D_{\text{score}}(i, u)$ 函数最小化目标函数。

$$\delta^* = \max_{\delta} \sum_{n=1}^N \left(E_{i \sim p_{\text{true}}(i|u_n, r)} \right) \left[\log D(i|u_n) \right] + E_{i \sim P_\theta(i|u_n, r)} \left[\log(1 - D(i|u_n)) \right], \quad (9)$$

其中 $p_{\text{true}}(i|u_n, r)$ 是潜在相关分布, 表示真实概率; $P_\theta(i|u_n, r)$ 是生成器 2 产生的估计概率。通过模型之间的相互作用, 可得到最优 δ 。判别器利用 RankNet^[23] 作为底层模型, 将 Pairloss 作为其损失函数。

$$\log D(i|u_n) = \text{Pairloss} = \sum_{n=1}^K \log \sigma \left(f_\delta(\text{pos}_n) - f_\delta(\text{neg}_n) \right). \quad (10)$$

训练流程如表 2 所示。

表 2 McRecGAN 的训练流程
Table 2 The training process of McRecGAN

Algorithm: MRecGAN
Require: The level one generator $p_\gamma(i u_n, r)$ and $g_{\text{score}}(i, u)$; The level two generator $p_\theta(i u_n, r)$ and $g_{\text{score}}(i, u)$; Discriminator $d_{\text{score}}(i, u)$ Dataset $S = \{ \text{Movielens100k} \}$ Initialize g_1, g_2 and d with random weights γ, θ and δ , respectively.
1: repeat
The generator cycles
2: Train the level one generator via the policy gradient (<i>ListLoss</i> * <i>Reward</i>)
3: $p_\gamma(i u_n, r)$ generates hard items for the level two generator
4: Train the level two generator via policy gradient using examples from $p_\gamma(i u_n, r)$
5: repeat
The generator-discriminator cycles
6: Train the level two generator via the policy gradient (<i>PointLoss</i> * <i>Reward</i>) using examples from $p_\gamma(i u_n, r)$
7: $p_\theta(i u_n, r)$ generates hard item pairs for the discriminator
8: Train the discriminator $d_{\text{score}}(i, u)$ using the <i>Pairloss</i>
9: Until MRecGAN converges

MRecGAN 使用策略梯度方法训练两个生成器与判别器。生成器涉及样本采样和梯度计算, 判别器会对生成器生成的样本和真实样本进行比较, 每个步骤的时间复杂度是 $O(d)$, 其中 d 是输入数据的维度, 所以模型训练的时间复杂度为 $O(td)$, 其中 t 是训练步数。同时, MRecGAN 需要处理和生成大量的数据, 并且生成器与判别器都基于神经网络, 那么这个过程的时间复杂度接近于 $O(d)$ 。另外, 处理和传输数据的时间复杂度通常与数据的大小成正比。综上, 可以近似地将 MRecGAN 算法的总计算复杂度表示为 $O(td)$ 。

3 实验结果与分析

3.1 推荐数据集

本文选取 Movielens (100 k) 推荐数据集, 该数据集包含了 943 位用户对 1 683 部、18 种类型的电影评分, 评分范围为 1 至 5。利用 2.2 中的方法可以将该推荐数据集转化为排序学习推荐数据集, 作为本算法使用的辅助信息。

3.2 效用评价准则

在推荐算法中,通常有两种效用评价准则,一种为预测的推荐列表中项目的得分与实际用户给项目的打分的偏差,另一种考虑了推荐项目的先后顺序。在实际应用中,推荐结果往往是有序的,用户偏好越相关的物品排序越靠前越好。因此,本文采用MAP(Mean Average Precision)、NDCG与MRR作为评价指标^[22]。

3.2.1 MAP

对于所有用户的平均准确率取平均值即为MAP,MAP值越大,说明用户偏好相关的物品排序越靠前,整体性能越好。

$$AP(u) = \frac{\sum_{k=1}^{n_u} P@k(u)}{|R|}, \quad (11)$$

$$P@k(u) = \frac{R(u) \cap T(u)@k}{R(u)@k} l_k, \quad (12)$$

$$MAP = \frac{\sum_u AP(u)}{|U|}, \quad (13)$$

其中 R 为用户 u 推荐列表的长度, n_u 为用户 u 推荐列表中的所有物品, l_k 为位置 k 处物品与用户的偏好是否相关:相关为1,不相关为0。

3.2.2 NDCG

对于目标用户 u ,其推荐列表排在位置 k 的物品DCG为:

$$DCG@k(u) = \sum_{i=1}^k G(u,i) D(i), \quad (14)$$

对所有用户求平均值,即为NDCG:

$$NDCG@k(u) = \frac{DCG@k(u)}{DCG^*@k(u)}, \quad (15)$$

其中 $G(u,i)$ 为增益函数,可以反映用户与物品偏好的相关度等级; $D(i)$ 为位置衰减函数,当位置越靠后衰减越大; DCG^* 为理想的DCG,即评分高的物品全部都排在了评分低的物品前面。同样地,NDCG值越大,说明排序性能越好。

3.2.3 MRR

MRR通过正确推荐项目在推荐结果中的排名来评估推荐算法的性能,其中 $|Q|$ 代表用户的个数, k_i 代表对于第 i 个用户,推荐列表中该项目在推荐项目中排列的位置。

$$MRR = \frac{1}{Q} \sum_{i=1}^{|Q|} \frac{1}{k_i}. \quad (16)$$

3.3 实验结果

3.3.1 基础模型训练情况

首先,对四种基础模型进行训练,并获取最佳性能下各自的评分数据。Autorec经过167次训练,最终RMSE(Root Mean Square Error)稳定在1.008附近,MAE(Mean Absolute Error)稳定在0.796附近;MF经过250次训练,最终RMSE稳定在1.133附近,MAE稳定在0.873附近;RNN经过500次训练,最终Precision稳定在0.6651,Recall稳定在0.8698,Accuracy稳定在0.6332;NMF经过250次训练,最终RMSE稳定在1.103附近,MAE稳定在0.858附近。图4是四种基础模型的性能指标与训练情况。

3.3.2 MRecGAN训练情况

数据输入到一个两层的神经网络中。第一层线性层由tanh作为激活函数,第二层由sigmoid函数激活。用于建模数据分布的生成概率由softmax函数计算。我们将生成器1的优化器设置为学习率为 1×10^{-6} 的SGD,将生成器2和判别器的优化器设置为学习率为 1×10^{-5} 的Adam,betas为(0.9,0.999),不使用权重衰减。

经过反复实验,得到了MRecGAN的最佳性能指标,算法每迭代十次,记录一次算法在测试集上的表现,其训练次数与性能变化情况如图5所示。

可以发现,模型的性能指标随着训练次数的增加而不断提高。在前60次的训练中,NDCG@3、NDCG@5、NDCG@10与MRR在测试集中提升较为迅速,分别达到0.6033、0.5148、0.4136与0.7081;在后100次的训练中,这些指标增长较为缓慢。MAP在前20次的训练中即达到了0.2141,后续缓慢增长。当算法迭代至性能无明显提高时,即可停止训练。经测试集检验,发现MRecGAN的NDCG@3达到0.8195,NDCG@5达到0.7601,NDCG@10达到0.6143,MAP达到0.2793,MRR达到0.8731。

3.3.3 消融实验

为了进一步探究生成器1对生成器2的辅助效果以及混合框架中特征维数对模型的影响情况,本文设置了两组消融实验,分别为生成器消融实验与特征消融实验。对于生成器消融,将生成器1去除(记作MRecGAN-outer),生

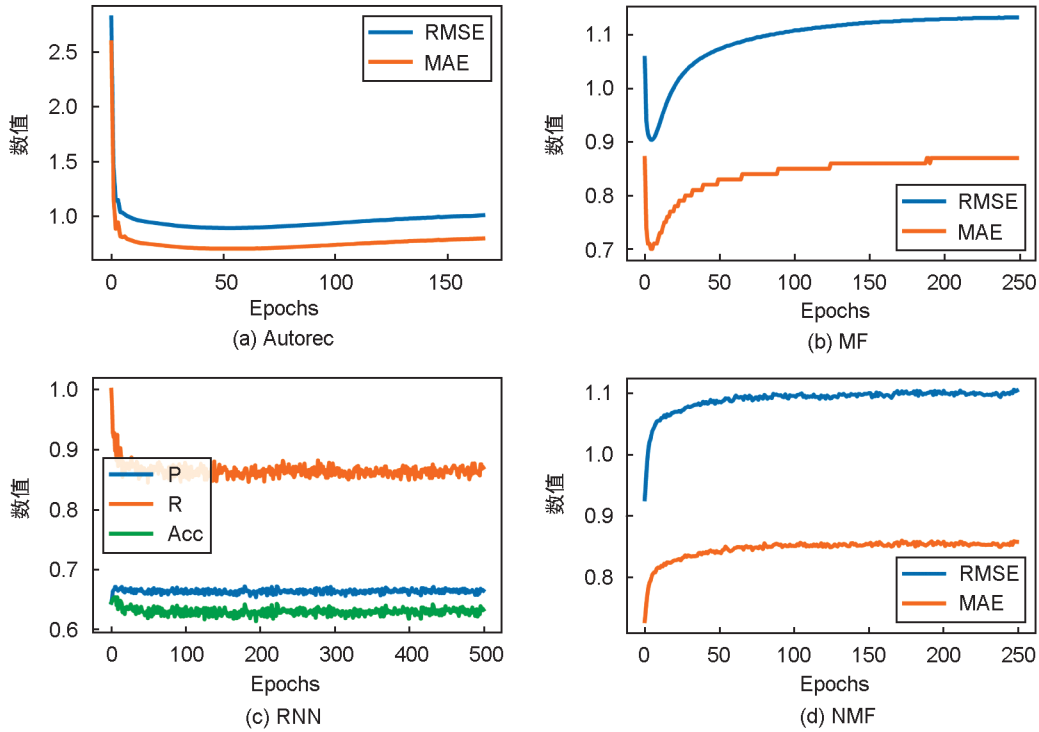


图4 基础模型性能情况

(a) Autorec; (b) MF; (c) RNN; (d) NMF

注:(a),(b),(d)采用RMSE与MAE作为评价指标,(c)采用P(Precision),R(Recall)与Acc(Accuracy)作为评价指标。

Fig. 4 Performance of the base model

(a) Autorec; (b) MF; (c) RNN; (d) NMF

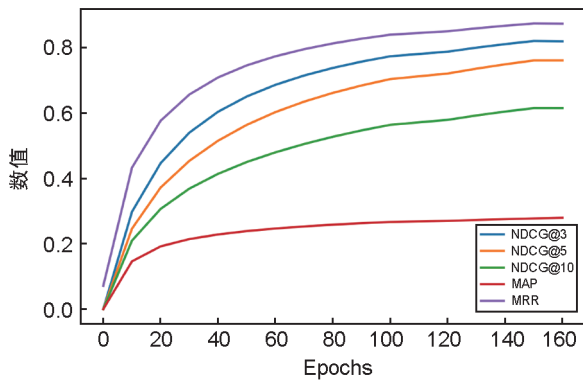


图5 MRecGAN训练情况

注:MRecGAN在训练过程中,NDCG@3,NDCG@5,NDCG@10,MAP与MRR指标不断升高,直至平稳。

Fig. 5 Training status of MRecGAN

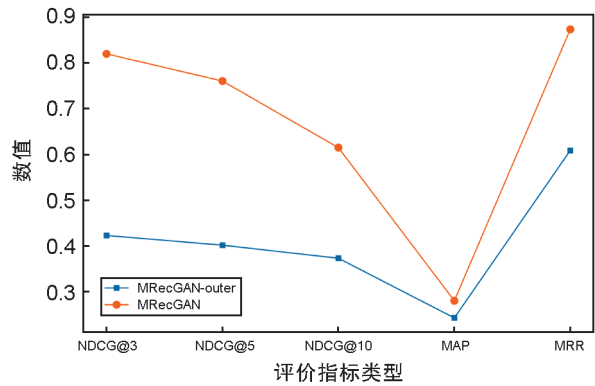


图6 MRecGAN分类器消融实验结果

注:MRecGAN-outer将生成器1去除,保留生成器2与判别器,以证明生成器1所起的作用。

Fig. 6 Results of the classifier ablation experiment in MRecGAN

成器2直接从排序学习推荐数据集中采样并与判别器展开对抗。在没有生成器1辅助的情况下,模型指标除MAP外均发生了大幅度地下降,可见生成器1对生成器2的训练起到了极大的帮助,它可以帮助生成器2进行采样,对模型整体性能的提升同样起到了至关重要的作用。消融实验结果如图6所示。

同时,针对NDCG指标设置特征消融实验。通过改变辅助信息的特征维数,可以挖掘出该模型辅助信息的最佳选取情况。当模型辅助信息过少时,即特征维数过少时,模型从辅助数据中挖掘的信息较少,推荐准确性较低;当模型辅助信息过多时,特征之间可能线性相关,导致训练难度加大的同时有效信息较少,同样

会降低推荐准确性。由实验结果可知,当特征维数在40左右时,该模型表现出最佳性能,实验结果如图7所示。

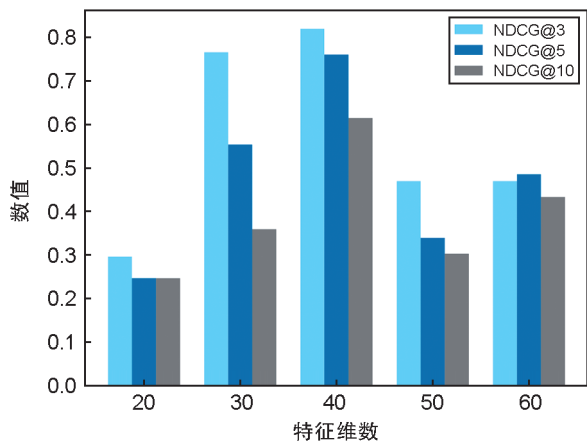


图7 MRecGAN特征消融实验结果

注:辅助信息的特征维数会影响推荐结果,分别设置维数(横轴)为20,30,40,50,60,采用(纵轴)NDCG@3,NDCG@5,NDCG@10作为指标,以探究最佳维数。

Fig. 7 Results of the feature ablation experiment on MRecGAN

通过改变基础模型的超参数,可以获得同一个基础模型下不同的评分特征。选取不同数量的基础模型评分特征,即可得到不同维数的模型整体特征。以特征维数40为例,该数据分别由四个基础模型的10个特征组成。特征的具体来源如表3所示。

表3 特征的具体来源

Table 3 Specific origins of features

特征维数	AutoRec	MF	NFM	RNN-Rec
20	5	5	5	5
30	8	8	7	7
40	10	10	10	10
50	13	13	12	12
60	15	15	15	15

3.3.4 模型对比

本文将 MRecGAN 与 CDAE (Collaborative Denoising Auto-encoders)^[24]、NCF (Neural Collaborative Filtering)^[25]、CML^[26]、LRML (Latent Relational Metric Learning)^[27]、IRGAN (A Minimax Game for Unifying Generative and Discriminative Information Retrieval Models)^[28]、BRP (Bayesian Personalized Ranking)^[29]、LightGCN (Simplifying and Powering Graph Convolution Network)^[30]、PD-GAN (Adversarial Learning for Per-

sonalized Diversity-promoting Recommendation)^[31]模型在 Movielens (100 k) 上的性能指标进行对比,得到表4。

表4 不同模型在 Movielens (100 k) 上的性能对比

Table 4 Performance comparison of different models on Movielens (100 k)

模型名称	NDCG@3	NDCG@5	NDCG@10	MAP	MRR
CDAE	0.216 9	0.211 5	0.204 7	0.137 4	0.392 3
NCF	0.445 8	0.418 3	0.391 2	0.252 7	0.622 9
CML	0.479 7	0.450 8	0.419 8	0.281 3	0.661 2
LRML	0.304 0	0.282 0	0.263 9	0.172 4	0.496 7
IRGAN	0.422 2	0.400 9	0.372 3	0.241 8	0.608 2
BRP	0.407 8	0.388 9	0.363 2	0.237 6	0.595 2
LightGCN	0.458 2	0.432 9	0.424 1	0.283 3	0.640 3
PD-GAN	0.429 4	0.481 6	0.455 0	—	—
MRecGAN	0.819 5	0.760 1	0.614 3	0.279 3	0.873 1
Improve/%	70.84	57.83	35.01	-1.41	32.05

注:Improve代表MRecGAN在同类指标下与最佳对比模型的对比结果。

值得注意的是,MRecGAN通过四种基础模型特征融合得到。这四种基础模型已经取得了较为优秀的推荐结果,本文将其推荐结果作为混合框架的辅助信息,会进一步提升算法的性能。同时,模型在NDCG与MRR方面提升明显,但MAP并没有显著提升,这是因为模型更加关注头部项目的排序质量,能够尽可能为用户推荐最相关的项目,并尽可能将最相关的项目排在前面,故而头部性能较好,而全部项目的整体排序性能提升不明显。另外,本文将 Movielens (100 k) 原始推荐数据集融入辅助信息,转化为了表现优秀的排序学习推荐数据集,故可以在排序方面表现较为优秀。

4 结束语

本文提出了一种基于对抗型排序学习的混合推荐算法(MRecGAN)。该算法将排序学习作为底层模型,通过多种推荐算法的特征融合,构建辅助信息,将推荐数据集转化为排序学习推荐数据集;同时,在对抗网络中新引入了一个生成器,辅助另一个生成器进行训练;生成器与判别器之间通过对抗的方式对排序学习函数进行学习,得到最终的推荐结果。MRecGAN吸取了不同推荐算法的优点,并且相互弥补了缺点,在 Movielens (100 k) 数据集上取

得了不错的效果。MRecGAN不仅能够有效缓解数据稀疏的问题,还为基于深度学习的混合推荐提出了一种统一的框架,为后续的研究提供的思路。当然,由于对抗网络本身的缺点,该算法的训练容易陷入局部极值,模型较为复杂,且稳定性不足。在未来的工作中,如何提高模型的稳定性,降低模型的复杂度将是进一步的研究内容。同时,本文目前只使用了Movielens(100 k),后续将会继续补充更多的数据集,进一步验证模型的泛化能力与可靠性。

此外,本文所提出的方法,同样适用于学术、商品与音乐等推荐场景。但相对于电影推荐,其他应用领域存在领域之间相关性不同、情感差异、评分刻度差异等问题。为了实现迁移学习,可以提取用户和项目的特征标签,把非结构化的信息转化为结构化数据,同时把特征进行量化与归一化处理,再迁移到目标任务中。

参考文献:

- [1] 郭艳红, 邓贵仕. 协同过滤系统项目冷启动的混合推荐算法[J]. 计算机工程, 2008, **34**(23): 11-13. DOI: 10.3969/j.issn.1000-3428.2008.23.005.
GUO Y H, DENG G S. Hybrid Recommendation Algorithm of Item Cold-start in Collaborative Filtering System[J]. *Comput Eng*, 2008, **34**(23): 11-13. DOI: 10.3969/j.issn.1000-3428.2008.23.005.
- [2] 吕成成, 王维国, 丁永健. 基于KNN-SVM的混合协同过滤推荐算法[J]. 计算机应用研究, 2012, **29**(5): 1707-1709. DOI: 10.3969/j.issn.1001-3695.2012.05.027.
LV C S, WANG W G, DING Y J. Hybrid Collaborative Filtering Algorithm Based on KNN-SVM[J]. *Appl Res Comput*, 2012, **29**(5): 1707-1709. DOI: 10.3969/j.issn.1001-3695.2012.05.027.
- [3] 冀振燕, 皮怀雨, 姚伟娜. 融合多源异构数据的混合推荐模型[J]. 北京邮电大学学报, 2019, **42**(1): 126-132. DOI: 10.13190/j.jbupt.2018-176.
JI Z Y, PI H Y, YAO W N. A Hybrid Recommendation Model Based on Fusion of Multi-source Heterogeneous Data[J]. *J Beijing Univ Posts Telecommun*, 2019, **42**(1): 126-132. DOI: 10.13190/j.jbupt.2018-176.
- [4] 胡思才, 孙界平, 琚生根, 等. 基于深度神经网络和概率矩阵分解的混合推荐算法[J]. 四川大学学报(自然科学版), 2019, **56**(6): 1033-1041. DOI: 10.3969/j.issn.0490-6756.2019.06.008.
HU S C, SUN J P, JU S G, et al. Hybrid Recommendation Algorithm Based on Deep Neural Network and Probabilistic Matrix Factorization[J]. *J Sichuan Univ Nat Sci Ed*, 2019, **56**(6): 1033-1041. DOI: 10.3969/j.issn.0490-6756.2019.06.008.
- [5] DARBAN Z Z, VALIPOUR M H. GHRs: Graph-based Hybrid Recommendation System with Application to Movie Recommendation[J]. *Expert Syst Appl*, 2022, **200**: 116850. DOI: 10.1016/j.eswa.2022.116850.
- [6] KARN A L, KARNA R K, KONDAMUDI B R, et al. Customer Centric Hybrid Recommendation System for E-commerce Applications by Integrating Hybrid Sentiment Analysis[J]. *Electron Commer Res*, 2023, **23**(1): 279-314. DOI: 10.1007/s10660-022-09630-z.
- [7] 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): 1619-1647. DOI: 10.11897/SP.J.1016.2018.01619.
HUANG L W, JIANG B T, LÜ S Y, et al. Survey on Deep Learning Based Recommender Systems[J]. *Chin J Comput*, 2018, **41**(7): 1619-1647. DOI: 10.11897/SP.J.1016.2018.01619.
- [8] 杨博, 赵鹏飞. 推荐算法综述[J]. 山西大学学报(自然科学版), 2011, **34**(3): 337-350. DOI: 10.13451/j.cnki.shanxi.univ(nat.sci.).2011.03.001.
YANG B, ZHAO P F. Review of the Art of Recommendation Algorithms[J]. *J Shanxi Univ Nat Sci Ed*, 2011, **34**(3): 337-350. DOI: 10.13451/j.cnki.shanxi.univ(nat.sci.).2011.03.001.
- [9] 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. New York: ACM, 2015: 111-112. DOI: 10.1145/2740908.2742726.
- [10] KOREN Y, BELL R, VOLINSKY C. Matrix Factorization Techniques for Recommender Systems[J]. *Computer*, 2009, **42**(8): 30-37. DOI: 10.1109/MC.2009.263.
- [11] HE X N, CHUA T S. Neural Factorization Machines for Sparse Predictive Analytics[C]// Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM, 2017: 355-364. DOI: 10.1145/3077136.3080777.
- [12] CHO K, VAN MERRIENBOER B, GULCEHRE C, et al. Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation[C]// Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Stroudsburg, PA, USA: Association for Computational Linguistics, 2014: 1724-1734. DOI: 10.3115/v1/d14-1179.
- [13] KO H, LEE S, PARK Y, et al. A Survey of Recommen-

- dation Systems: Recommendation Models, Techniques, and Application Fields[J]. *Electronics*, 2022, **11**(1): 141. DOI: 10.3390/electronics11010141.
- [14] GOODFELLOW I, POUGET-ABADIE J, MIRZA M, *et al.* Generative Adversarial Networks[J]. *Commun ACM*, 2020, **63**(11): 139–144. DOI: 10.1145/3422622.
- [15] CHEN H, WANG Z F, HUANG F R, *et al.* Generative Adversarial Framework for Cold-start Item Recommendation[C]//Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM, 2022: 2565–2571. DOI: 10.1145/3477495.3531897.
- [16] REN R Y, LIU Z Y, LI Y L, *et al.* Sequential Recommendation with Self-attentive Multi-adversarial Network[C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM, 2020: 89–98. DOI: 10.1145/3397271.3401111.
- [17] LI C L, ZHAO M J, ZHANG H M, *et al.* RecGURU: Adversarial Learning of Generalized User Representations for Cross-domain Recommendation[C]//Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining. New York: ACM, 2022: 571–581. DOI: 10.1145/3488560.3498388.
- [18] 林原. 信息检索中排序学习方法的研究[D]. 大连: 大连理工大学, 2012.
LIN Y. Research of Learning to Rank in Information Retrieval[D]. Dalian: Dalian University of Technology, 2012.
- [19] XU B, LIN H F, YANG L, *et al.* Cognitive Knowledge-aware Social Recommendation via Group-enhanced Ranking Model[J]. *Cogn Comput*, 2022, **14**(3): 1055–1067. DOI: 10.1007/s12559-022-10001-x.
- [20] ZHU N J, CAO J, LU X J, *et al.* Leveraging Pointwise Prediction with Learning to Rank for Top-N Recommendation[J]. *World Wide Web*, 2021, **24**(1): 375–396. DOI: 10.1007/s11280-020-00846-3.
- [21] 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. New York: ACM, 2020: 528–536. DOI: 10.1145/3336191.3371831.
- [22] 黄震华, 张佳雯, 田春岐, 等. 基于排序学习的推荐算法研究综述[J]. *软件学报*, 2016, **27**(3): 691–713. DOI: 10.13328/j.cnki.jos.004948.
- HUANG Z H, ZHANG J W, TIAN C Q, *et al.* Survey on Learning-to-rank Based Recommendation Algorithms[J]. *J Softw*, 2016, **27**(3): 691–713. DOI: 10.13328/j.cnki.jos.004948.
- [23] BURGESS C. From RankNet to LambdaRank to LambdaMART: An Overview[J]. *Learning*, 2010, **11**(23-581):81.
- [24] 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. New York: ACM, 2016: 153–162. DOI: 10.1145/2835776.2835837.
- [25] 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. New York: ACM, 2017: 173–182. DOI: 10.1145/3038912.3052569.
- [26] HSIEH C K, YANG L Q, CUI Y, *et al.* Collaborative Metric Learning[C]//Proceedings of the 26th International Conference on World Wide Web. New York: ACM, 2017: 193–201. DOI: 10.1145/3038912.3052639.
- [27] TAY Y, TUAN L A, HUI S C. Latent Relational Metric Learning via Memory-based Attention for Collaborative Ranking[C]//Proceedings of the 2018 World Wide Web Conference. New York: ACM, 2018: 729–739. DOI: 10.1145/3178876.3186154.
- [28] WANG J, YU L T, ZHANG W N, *et al.* IRGAN: a Minimax Game for Unifying Generative and Discriminative Information Retrieval Models[C]//Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM, 2017: 515–524. DOI: 10.1145/3077136.3080786.
- [29] RENDLE S, FREUDENTHALER C, GANTNER Z, *et al.* BPR: Bayesian Personalized Ranking from Implicit Feedback[C]//Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence. New York: ACM, 2009: 452–461. DOI: 10.5555/1795114.1795167.
- [30] HE X N, DENG K, WANG X, *et al.* LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation[C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM, 2020: 639–648. DOI: 10.1145/3397271.3401063.
- [31] WU Q, LIU Y, MIAO C Y, *et al.* PD-GAN: Adversarial Learning for Personalized Diversity-promoting Recommendation[C]//Proceedings of the 28th International Joint Conference on Artificial Intelligence. New York: ACM, 2019: 3870–3876. DOI: 10.5555/3367471.3367579.