

面向隐式反馈的标签感知推荐方法

李红梅¹ 刁兴春¹ 曹建军² 冯 钦¹ 张 磊¹

(陆军工程大学 南京 210007)¹ (国防科技大学第六十三研究所 南京 210007)²

摘要 为进一步提高面向隐式反馈的标签感知推荐性能,针对隐式反馈数据的稀疏性问题以及标签数据的冗余、语义模糊等问题,提出了一种基于用户细粒度偏好和增量加权矩阵分解的个性化推荐方法。为缓解隐式反馈数据稀疏不平衡的影响,提出使用协同近邻用户关系从大规模未观测数据中挖掘目标用户可能感兴趣的潜在项目,即近邻用户感兴趣但目标用户未选择的项目,进而提出了用户对项目的细粒度偏好假设:观测项目>潜在项目>其他未观测项目,改进传统成对偏好假设的粗糙性。为获取更为可靠的近邻用户,利用基于深度学习的方法来抽取用户-标签的低维、抽象的深层语义特征,缓解了原始标签数据的冗余、语义模糊等对用户表征的影响。最后,基于用户的细粒度偏好提出一种增量加权矩阵分解模型,并进行快速优化求解与推荐。实验结果表明:提出的算法在多个排序推荐准确性的评价指标(Pre@5, NDCG@5, MRR)上分别提升了约 9%, 8%, 9%, 验证了所提算法的有效性。

关键词 隐式反馈, 标签感知推荐, 深度学习, 细粒度偏好, 加权矩阵分解

中图分类号 TP391.3 文献标识码 A DOI 10.11896/j.issn.1002-137X.2019.04.006

Tag-aware Recommendation Method with Implicit Feedback

LI Hong-mei¹ DIAO Xing-chun¹ CAO Jian-jun² FENG Qin¹ ZHANG Lei¹

(Army Engineering University of PLA, Nanjing 210007, China)¹

(The 63rd Research Institute, National University of Defense Technology, Nanjing 210007, China)²

Abstract In order to further improve the performance of tag-aware personalized recommendation with implicit feedbacks, aiming at the problems of the redundancy, ambiguity of tagging information and the sparsity and imbalance of implicit feedbacks, this paper proposed a personalized recommendation method based on fine-grained preference assumption and augmented weighted matrix factorization. First, one kind of candidate items that the target user may prefer are mined by leveraging its neighbor user, which are preferred by neighbor users which have not been selected by the target user. Thus, a type of fine-grained preference relationship among three kinds of items for target users is obtained, i. e., observed item>candidate item>other unobserved data. This kind of operation can help to alleviate the sparsity and imbalance problem. Then, the deep learning method is used to extract the in-depth semantic features from tag space. In this way, representations of users' profiles become more abstract and advanced, and user neighbors are obtained based on the in-depth semantic features. Afterwards, a revised weighted matrix factorization model is formulated based on the fine-grained preference relationship for personalized recommendation. And a fast eALS algorithm is used for model optimization in terms of low time complexity. Experiments on real-world datasets show that the proposed method outperforms competing methods on several evaluation metrics, including Pre@5, NDCG@5, MRR. The three indicators are respectively increased by 9%, 8%, and 9%, which indicates the effectiveness of the proposed methods.

Keywords Implicit feedback, Tag-aware recommendation, Deep learning, Fine-grained preference, Weighted matrix factorization

1 引言

推荐系统能够根据不同用户的历史行为信息,帮助用户从海量信息中快速发现其可能感兴趣的信息,并以推荐列表的形式推送给用户,又称 top-n 个性化推荐。这些历史行为信息主要包括显式反馈(例如评分、等级)和隐式反馈(例如点击、浏览、收藏)。相对于显式反馈(Explicit Feedback),隐式

反馈(Implicit Feedback)没有显式的偏好等级或者负类反馈,仅包含用户的单类正反馈信息,且无需用户主动提供,在实际应用场景中更加普遍且容易获取。近些年,面向隐式反馈的个性化推荐得到越来越多的关注与研究^[1-4]。

但隐式反馈数据的稀疏性问题给传统的推荐方法带来了一定的挑战。这是因为通常大部分用户仅选择了少数资源(统称项目),其规模远远小于其他未选择的项目;同时,用户

收稿日期:2018-08-15 返修日期:2018-11-11 本文受国家自然科学基金面上项目(61371196)资助。

李红梅(1990-),女,博士生,主要研究方向为个性化推荐;刁兴春(1964-),男,研究员,博士生导师,主要研究方向为数据工程;曹建军(1975-),男,副研究员,硕士生导师,主要研究方向为数据智能分析与应用、数据质量控制与数据治理, E-mail: xinxizhiliang@163.com(通信作者);冯钦(1993-),男,硕士生,主要研究方向为数据工程;张磊(1989-),男,硕士生,主要研究方向为数据工程。

与项目的隐式交互行为无法反映用户的偏好内容,在一定程度上影响了推荐的准确性。

随着 Web 技术的发展,越来越多的信息服务系统允许用户为资源自由地添加标签(关键字),例如音乐推荐系统 Last.fm、电影推荐系统 MovieLens、文献推荐系统 Citeulike、书签推荐系统 Delicious、商品推荐系统 Amazon、微博等。这些标签很好地表达了项目的特征及用户的偏好内容,又涵盖了用户与资源之间、用户与用户之间的关系,兼具内容与关联的特征,且非常容易获取。将标签作为传统隐式反馈数据的补充,有可能开发出兼具内容过滤和协同过滤的优越性的推荐技术,以弥补隐式反馈数据在偏好内容表征上的不足,同时在一定程度上缓解数据的稀疏性问题。

很多学者提出利用标签信息来辅助提高面向隐式反馈的协同过滤推荐,该方法又称标签感知的推荐(Tag-aware Recommendation),其主要分为两类:基于近邻的方法^[5-6]和基于模型的方法^[7-8]。基于近邻的标签感知推荐方法主要基于传统的 k 近邻方法,包括基于用户近邻^[9]和基于项目近邻的方法^[10],其基于标签数据计算用户相似度或项目相似度,进而生成推荐。文献^[11]充分考虑了用户相似度和项目相似度的有效融合,进一步提高了推荐性能。基于模型的标签感知推荐方法通过将用户-标签或项目-标签矩阵与传统的模型方法相结合,并对模型参数进行优化求解来生成推荐。常用的模型包括矩阵分解模型^[12-13]、扩散模型^[7]、随机游走模型^[14]、主题模型^[15]、深度学习模型^[6]等。

然而,现有的方法存在两方面的缺陷:1)传统基于成对偏好假设的方法对稀疏隐式反馈数据的适用性降低,忽略了对大规模未观测数据的深入挖掘,在一定程度上影响了对用户偏好的准确刻画;2)标签数据的冗余、语义模糊等问题也对基于标签的用户表征产生了影响,降低了标签感知推荐的性能。

针对以上问题,文中提出了一种基于用户细粒度偏好和加权矩阵分解的个性化推荐方法。首先,利用用户近邻关系从大规模未观测项目中挖掘目标用户可能感兴趣的潜在项目,即目标用户未选择但近邻用户选择的项目,提出一种用户对项目的细粒度偏好假设(观测项目>潜在项目>其他未观测项目),缓解隐式反馈数据的稀疏性的影响;同时,为获取更为可靠的近邻用户集合,针对原始标签数据的冗余、语义模糊等问题,利用深度学习技术逐层、无监督地学习用户-标签的深层语义特征,并基于该深层语义特征计算用户间的相似度,获取与目标用户兴趣相似的近邻用户集合;最后,提出一种增量的加权矩阵分解模型对细粒度偏好关系进行建模,构建个性化推荐模型,并对模型参数进行快速求解,完成对未观测项目的评分预测、排序与推荐。所提方法的流程如图 1 所示。

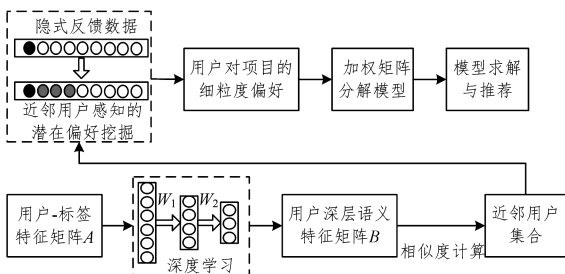


图 1 面向稀疏隐式反馈的标签感知推荐流程图

Fig. 1 Procedure of tag-aware recommendation with sparse implicit feedback

2 问题定义与数据分析

2.1 问题定义

已知用户集合 $U = \{u_1, \dots, u_M\}$ 、项目集合 $\mathcal{I} = \{i_1, \dots, i_N\}$ 和标签集合 $\mathcal{G} = \{g_1, \dots, g_Z\}$ 。给定 M 个用户对 N 个项目的隐式反馈数据集 $\mathcal{S} = \{(u, i) | u \in U, i \in \mathcal{I}\}$ 以及用户对项目的标注数据 $\mathcal{L}_u = \{(u, i, g) | (u, i) \in \mathcal{S}, g \in \mathcal{G}\}$, 目的是利用标注数据辅助隐式反馈进行推荐, 为每个用户推荐个性化的项目排序列表。

2.2 数据分析

以文献^[6]中的 Lastfm 数据集为例, 通过对数据集进行预处理, 过滤掉出现频次少于 5 的标签, 得到稀疏度为 0.3% 的数据集, 同时分别对每个用户标注的项目个数进行统计(项目总数为 12212), 得到用户访问项目频次统计图, 如图 2 所示, 横坐标表示项目个数, 纵坐标表示访问不同项目个数的用户数量, 例如, 访问项目个数为 0~10 之间的用户数量高达 1000。

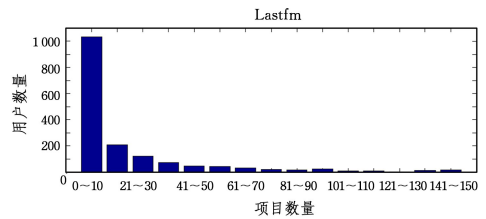


图 2 用户-项目访问频次统计图

Fig. 2 User-item visit frequency

从图 2 可以看出, 绝大多数用户仅对少数项目(频次小于或等于 10, 比例小于等于 0.08%)感兴趣, 可见用户的观测项目远远小于未观测项目, 这种稀疏不平衡特性对传统的协同过滤推荐算法(例如矩阵分解、 k -近邻、贝叶斯个性化排序等)带来一定的挑战, 容易造成结果偏置、过拟合等问题, 影响推荐的准确性。

事实上, 通常认为用户的选择行为受到社交(朋友)的影响, 用户很有可能对朋友选择的项目感兴趣, 即用户的大量未观测数据中隐藏着部分用户可能感兴趣的项目。通过挖掘这些用户潜在感兴趣的项目, 可得到用户对项目的细粒度偏好关系, 进而结合传统的协同过滤推荐模型进行用户偏好建模与推荐。该思路可辅助挖掘用户的潜在偏好信息, 缓解数据稀疏不平衡的影响, 提高用户偏好建模的准确性。鉴于此, 本文提出利用标签数据获取内容感知的近邻用户, 从近邻用户中挖掘用户潜在感兴趣的项目, 进而对用户的细粒度偏好进行建模与推荐。

3 标签感知的用户细粒度偏好挖掘

首先, 利用用户近邻关系定义每个用户对不同类型项目的细粒度偏好关系; 然后, 针对标签数据的冗余、语义模糊问题, 提出利用深度学习技术来表征用户-标签的深层语义特征, 进而根据深层语义特征计算近邻用户。

3.1 近邻感知的用户细粒度偏好假设

传统的推荐方法大多基于用户对观测项目和未观测项目的成对偏好假设进行建模, 将观测项目视为“正反馈”, 将所有的未观测项目视为“负反馈”, 认为用户对观测项目的偏好程度相对大于未观测项目, 如图 3(a)所示。其中“1”表示用户

的观测项目,空白部分即为未观测项目。基于近邻用户关系,本文提出一种近邻感知的细粒度偏好假设,即用户对3种不同类型项目的细粒度偏好序列,如图3(b)所示。

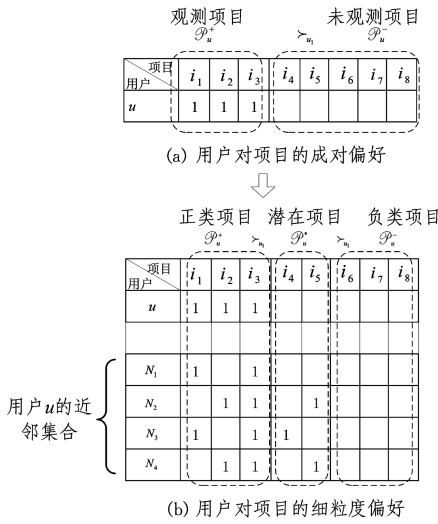


图3 用户的细粒度偏好示意图

Fig. 3 Fine-grained preference diagram of users on different items

定义1(正类项目 \mathcal{P}_u^+) 用户 u 选择的项目集合,表达用户对其感兴趣,例如用户浏览、点击、标注的项目集合。

定义2(潜在项目 \mathcal{P}_u^*) 用户 u 未选择但可能感兴趣的项目集合。例如,用户 u 的朋友或者兴趣相似的近邻用户 $\{u_1, u_2, \dots, u_{N_u}\}$ 所感兴趣但用户 u 未选择的项目。这是因为用户的选择行为很容易受到近邻用户的影响,用户对近邻用户选择的项目感兴趣的程度大于随机选择的项目。潜在项目 \mathcal{P}_u^* 可形式化为: $\mathcal{P}_u^* = \{\mathcal{P}_{u_1}^+ \cup \mathcal{P}_{u_2}^+ \cup \dots \cup \mathcal{P}_{u_{N_u}}^+\} \setminus \mathcal{P}_u^+$ 。

如图3(b)所示,对于目标用户 u 来说,假设其 N_u ($N_u = 4$) 个近邻用户集合为 $\mathcal{A}(u) = \{u_1, u_2, u_3, u_4\}$, 用户 u 感兴趣的项目集合为 $\mathcal{P}_u^+ = \{i_1, i_2, i_3\}$, 其近邻用户所感兴趣的项目集合为 $\mathcal{I}_{N_u} = \{\mathcal{P}_{u_1}^+ \cup \mathcal{P}_{u_2}^+ \cup \mathcal{P}_{u_3}^+ \cup \mathcal{P}_{u_4}^+\} = \{i_1, i_2, i_3, i_4, i_5\}$, 则可认为用户 u 可能对潜在项目 $\mathcal{P}_u^* = \mathcal{I}_{N_u} \setminus \mathcal{P}_u^+ = \{i_4, i_5\}$ 感兴趣。潜在项目的个数与目标用户 u 的近邻用户个数 N_u 有关。

定义3(负类项目 \mathcal{P}_u^-) 用户未选择且不太感兴趣的项目集合,例如用户 u 和近邻用户都未选择的项目集合。

基于以上定义,认为用户对正类项目的偏好程度大于潜在项目,对潜在项目的偏好程度大于其他未观测项目,即 $i^+ >_u i^* >_u i^-$ 。对每个用户 u 构建细粒度偏好序列样本集合 \mathcal{F} :

$$\mathcal{F} = \{(i^+, i^*, i^-) | i^+ \in \mathcal{P}_u^+, i^* \in \mathcal{P}_u^*, i^- \in \mathcal{P}_u^-\} \quad (1)$$

3.2 基于标签深层语义的近邻用户计算

合理确定潜在项目 \mathcal{P}_u^* 对推荐算法至关重要,其关键在于如何更好地确定近邻用户。其中,近邻用户获取的可靠性依赖于用户特征表示的准确程度。

常用的方法是直接基于用户-标签的 TF(Term Frequency, TF)或者 TF-IDF(Term Frequency-Inverse Document Frequency, TF-IDF)表示用户特征向量。但这种方法仅考虑标签出现的频次,不考虑标签的语义信息。事实上,作为表达用

户偏好内容和项目特征的一种自然语言,标签数据通常面临冗余、语义模糊等问题。直接利用原始标签数据计算用户的相似度,会在一定程度上影响相似度计算的准确性^[6]。

由于深度学习具有强大的从样本中学习数据集本质特征的能力,可通过学习一种深层次非线性网络结构来获取用户和项目的深层次特征表示,缓解数据的冗余、语义模糊等问题,其已被成功应用于推荐^[16]、网页搜索^[17]、文本、图像分类^[18]等多个领域。本文采用文献^[6]中的深层稀疏自编码器作为深度学习模型,通过将多个稀疏自编码器(Sparse Auto Encoder, SAE)堆叠起来形成的一种深层网络模型,自动从无标注数据中逐层、无监督地学习用户-标签的特征。

基本的自编码器通过一个编码和一个解码过程来重构输入数据,学习数据的隐层表示,通常可视为一个3层神经网络结构:一个输入层、一个隐层和一个输出层。自编码器的目的是使得输入与输出尽可能接近,这种接近程度通过重构误差(即损失函数)表示。但基本的自编码器很容易学习到一个恒等函数,为解决该问题,研究者提出了一系列自编码器的变种,例如稀疏自编码器,它通过添加一些限制性条件(比如稀疏性),在隐藏层得到输入数据的非线性表示。

假设模型的输入样本为 $\{a_1, a_2, \dots, a_M\}$, 本文中每个样本即为用户-标签权重向量,隐藏层可表示为:

$$h(a^i; W, b) = \sigma(a^i W + b) \quad (2)$$

其中, W 为输入层和隐藏层之间的权重矩阵; b 是输入层的偏置; σ 是一种激活函数,常见的激活函数包括 sigmoid, tanh 函数等。

从输入层到隐藏层的非线性映射过程又称编码过程,从隐藏层到输出层的非线性映射过程又称解码过程,即输入层数据在输出层的重构。由此,目标函数是最小化输入和输出之间的重构误差:

$$\min_{W, b, c} \sum_{i=1}^M \| \sigma(h(a^i; W, b) W^T + c) - a^i \| \quad (3)$$

其中, c 是隐藏层的偏置。

稀疏自编码器通过在损失函数中加入稀疏惩罚项来对过大的权重进行惩罚,使隐层表示中的大量节点为0,从而确保隐层表示尽量稀疏,提高了特征表达的能力^[18-19]。稀疏惩罚项 KL-散度(Kullback-Leibler, KL)用来约束原目标函数^[20]。最小化 KL-散度如下:

$$\min_{\rho} \sum_{j=1}^n D_{KL}(\rho \| \hat{\rho}_j) \quad (4)$$

KL-散度通常用于度量两个概率分布的差别,即最小化隐藏层的期望 $E[h(a^i; W, b)]$ 和平均激活值 ρ (ρ 通常是比较小的值,例如 $\rho = 0.1$), n 为隐藏层的单元个数。

这样,稀疏自编码器的目标函数形式化为:

$$\min_{W, b, c} \sum_{i=1}^M \| \sigma(h(a^i; W, b) W^T + c) - a^i \| + \lambda \sum_{j=1}^n D_{KL}(\rho \| \hat{\rho}_j) \quad (5)$$

其中, λ 表示稀疏惩罚项的系数。

在基础的自编码器上增加多个隐含层,即可构成深度自编码器,也称为栈式自编码器。模型的预训练采用逐层学习算法。在该算法中,首先将原始数据作为输入对第一个隐含

层参数进行训练,将第一个隐含层得到的激活值作为输入来训练第二个隐含层参数,重复该过程直到所有参数被训练完;然后,对整个网络参数进行微调,得到整个网络的参数;最后,根据训练好的网络得到所有用户的深层语义特征 $B \in \mathbb{R}^{N \times P}$ 。

基于深层语义特征获取目标用户 u 的前 N_u 个近邻用户集合 $\mathcal{A}(u)$ 。值得注意的是,近邻用户集合的大小 N_u 影响着推荐模型的性能,具体参数分析见实验部分(5.3.3节)。接下来,便可根据近邻用户集合 $\mathcal{A}(u)$ 的隐式反馈行为来确定潜在项目 \mathcal{P}_u^* ,进而构建目标用户 u 的细粒度偏好。

4 基于加权矩阵分解和细粒度偏好的推荐

4.1 推荐构建

传统的加权矩阵分解模型^[1-3,21-23]主要基于用户对项目的成对偏好进行建模。该方法将观测数据 \mathcal{J}_u^+ 视为正样例,将未观测数据 \mathcal{J}_u^- 视为负样例,分别对观测数据和未观测数据设定相应的置信权重^[1,2,24]。具体地,通过将观测数据的初始值设为 1、未观测数据的初始值设为 0,并分别设定不同的置信权重来构建误差损失函数,通过最小化观测数据和预测数据之间的加权损失,得到用户和项目的隐特征矩阵并进行评分预测。通常,为防止过拟合,在损失函数上添加 L2 正则项约束来构成目标函数:

$$\begin{aligned} L(U, V) &= \sum_{u=1}^M \sum_{i=1}^N C_{ui} (R_{ui} - \hat{R}_{ui})^2 + \lambda (\sum_{u=1}^M \|U_u\|^2 + \sum_{i=1}^N \|V_i\|^2) \\ &= \sum_{u=1}^M (\sum_{i^+ \in \mathcal{P}_u^+} C_{ui^+} (1 - R_{ui^+})^2 + \sum_{i^- \in \mathcal{P}_u^-} C_{ui^-} (0 - R_{ui^-})^2) + \lambda (\sum_{u=1}^M \|U_u\|^2 + \sum_{i=1}^N \|V_i\|^2) \quad (6) \end{aligned}$$

其中, $U \in \mathbb{R}^{M \times K}$ 和 $V \in \mathbb{R}^{N \times K}$ 分别表示用户和项目的隐特征矩阵,两个矩阵共享低维隐特征空间,分别刻画用户偏好和项目特性。两者的内积表示用户 u 对项目 i 的偏好程度 $\hat{R}_{ui} = \sum_{k=1}^K U_{uk} V_{ik}$ (K 的大小影响着推荐模型的性能,具体参数分析见,5.3.2节实验部分)。正则项因子 λ 用于防止过拟合。 C_{ui} 则表示用户 u 对项目 i 偏好的置信权重,通常将观测数据的置信权重统一设为 1,其他未观测数据的置信权重则小于 1:

$$C_{ui} = \begin{cases} 1, & i \in \mathcal{P}_u^+ \\ w, & i \notin \mathcal{P}_u^+ \end{cases} \quad (7)$$

不同于传统的加权矩阵分解模型,本文基于用户对不同类型项目的细粒度偏好来修正加权矩阵分解模型。具体地,通过分别对 3 类项目赋予不同的初始值,即正类项目 \mathcal{P}_u^+ 、潜在项目 \mathcal{P}_u^* 和其他未知项目 \mathcal{P}_u^- ,来表征用户对不同类型项目的不同偏好程度:

$$R_w = \begin{cases} 1, & i \in \mathcal{P}_u^+ \\ \alpha, & i \in \mathcal{P}_u^* \\ 0, & i \in \mathcal{P}_u^- \end{cases} \quad (8)$$

其中, $0 < \alpha < 1$ 。参数 α 的选择见 5.3.1 节实验部分。

相应地,可将目标函数转化为:

$$\begin{aligned} L(U, V) &= \sum_{u=1}^M (\sum_{i^+ \in \mathcal{P}_u^+} C_{ui^+} (1 - R_{ui^+})^2 + \sum_{i^* \in \mathcal{P}_u^*} C_{ui^*} (\alpha - R_{ui^*})^2 + \sum_{i^- \in \mathcal{P}_u^-} C_{ui^-} (0 - R_{ui^-})^2) + \\ &\quad \lambda (\sum_{u=1}^M \|U_u\|^2 + \sum_{i=1}^N \|V_i\|^2) \quad (9) \end{aligned}$$

4.2 模型学习

eALS 是一种元素级的 ALS 方法,该方法通过固定隐藏特征向量中其他维度的数值来优化每维坐标的数值。特别地,该方法能够避免 ALS 方法的矩阵逆运算,有效提高模型的优化速率。本文采用元素级的交替最小二乘(element-wise Alternative Least Square, eALS)方法对参数进行优化。

首先,对用户特征向量的每个元素 U_{uk} 进行求导,同时令

$$\hat{R}_{ui}^k = R_{ui} - U_{uk} V_{ik}, \text{且导数为 } 0。$$

$$\begin{aligned} \frac{\partial L}{\partial U_{uk}} &= \sum_{i=1}^N C_{ui} (R_{ui} - \hat{R}_{ui}^k) (-V_{ik}) + \lambda U_{uk} \\ &= \sum_{i=1}^N C_{ui} (R_{ui} - \hat{R}_{ui}^k - U_{uk} V_{ik}) (-V_{ik}) + \lambda U_{uk} \\ &= 0 \end{aligned} \quad (10)$$

然后,可得到 U_{uk} 闭合形式的优化公式:

$$U_{uk} = \frac{\sum_{i=1}^N C_{ui} (R_{ui} - \hat{R}_{ui}^k) V_{ik}}{\sum_{i=1}^N C_{ui} V_{ik}^2 + \lambda} \quad (11)$$

同样,可得到 V_{ik} 的闭合优化公式:

$$V_{ik} = \frac{\sum_{u=1}^M C_{ui} (R_{ui} - \hat{R}_{ui}^k) U_{uk}}{\sum_{u=1}^M C_{ui} U_{uk}^2 + \lambda} \quad (12)$$

4.3 模型学习

WRMF-Deep 算法的具体流程如算法 1 所示。

算法 1 WRMF-Deep

输入:用户-项目交互矩阵 R ,用户-标签权重矩阵 A ,用户和项目的隐特征维度 K ,正则项参数 λ ,最大迭代次数 γ

输出:模型参数 $\Theta = \{U, V\}$

1. 初始化:用户和项目的隐特征矩阵 $U \sim N(0, 0.01)$, $V \sim N(0, 0.01)$
2. 获取标签深层语义感知的近邻用户(3.2节)
3. 构建近邻用户感知的细粒度偏好集合(3.1节)
4. 根据每个用户对项目的细粒度偏好,设定加权矩阵分解模型的权重 w 和初始值 α (4.1节)
5. while 目标函数(式(9))未收敛并且迭代次数 \leq 最大迭代次数 γ
6. for $u=1, \dots, M$
7. for $k=1, \dots, K$
8. 根据式(11)更新用户的隐特征元素 U_{uk}
9. endfor
10. endfor
11. for $i_-=1, \dots, N$
12. for $k=1, \dots, K$
13. 根据式(12)更新每个项目的隐特征 V_{ik}
14. endfor
15. endfor
16. 返回 $\Theta = \{U, V\}$

得到参数 $\Theta = \{U, V\}$ 后,便可利用 $\hat{R}_{ui} = \sum_{k=1}^K U_{uk} V_{ik}$ 计算用户 u 对所有未观测项目 i 的偏好值,然后根据偏好值大小

确定每个用户的 top- n 个性化推荐列表。

复杂度分析:算法的复杂度主要依赖于两部分:1)根据近邻用户确定模型的参数(算法1的第2-4行),但该部分可以离线处理,不影响参数的在线训练时间;2)基于 eALS 优化,由于该算法避免了矩阵的逆运算,其计算复杂度为 $O(MNK^2)$ 。

5 实验结果与分析

5.1 数据集与度量标准

实验采用文献[6]的 HetRec2011 的 Lastfm 数据集¹⁾进行训练与测试,该数据集是用于推荐领域的标准数据集。Lastfm 数据集主要包含用户对音乐网站 Last.fm 中歌手的收听信息和用户对歌手添加的标签。该数据集包含了 1892 名用户对 12523 个歌手信息的 186479 条标注记录。首先对数据进行预处理,过滤掉出现频次少于 5 的标签。采用 5-交叉验证的方法训练与测试模型,构建训练集(80%)和测试集(20%),并取平均值作为最终结果。

采用 4 种推荐度量标准来衡量推荐效果,包括平均准确率均值(Mean of average precision, $Pre@n$)^[8]、归一化的累积折扣增益(Normalized Discounted Cumulative Gain, $NDCG@n$)^[6]、平均排序倒数(Mean Reciprocal Rank, MRR)^[8]。

5.2 对比方法与参数设置

本文选择以下几种算法与所提算法 WRMF-Deep 进行对比。

1)Pop。该方法基于项目的流行度对其进行排序推荐,认为访问的用户越多,该项目越流行,其在推荐列表中的排位越靠前。该方法是一种非个性化推荐,即对任何用户而言,所用项目的排序一致。

2)KNN。该方法是一种基于用户的协同过滤推荐方法,根据用户与项目的交互数据计算用户间的相似度,并根据近邻用户的相似度计算用户对未观测项目的评分,根据评分大小进行排序与推荐。

3)BPR。该方法是一种基于成对偏好假设的贝叶斯个性化排序推荐方法,其认为用户对观测数据的偏好程度大于未观测数据,并试图将每个用户感兴趣的观测项目排在其他未观测数据前面。为进行有效对比,本文采用矩阵分解技术和均匀抽样的策略训练模型^[25]。

4)WRMF。该方法是一种基于逐点偏好假设的加权矩阵分解方法,通常将隐式反馈中的观测数据置为 1,未观测数据置为 0,通过分别对观测数据和未观测数据添加相应的置信权重,利用逐点回归方法对用户偏好进行评分预测^[2]。

5)KNN-Tag。直接利用用户-标签表征用户,然后采用用户近邻的方法进行评分预测与推荐。

6)KNN-Deep。利用深度学习的方法来学习用户-标签的深层语义特征,然后采用用户近邻的方法进行评分预测与推荐,有效地缓解了标签数据的冗余、语义模糊等问题,进而提高了标签感知的推荐性能^[6]。

参数设置:将用户特征矩阵和项目特征矩阵初始化为均值为 0、方差为 0.01 的标准正态分布,分别为 $U \sim N(0, 0.01)$ 和 $V \sim N(0, 0.01)$,最大迭代次数 $\gamma = 100$ 。正则项系数范围

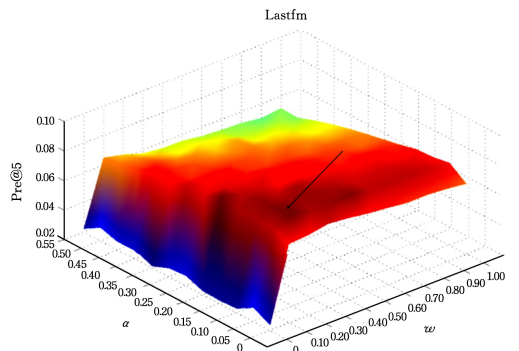
为 $\lambda_U = \lambda_V \in \{0.001, 0.01, 0.1, 1\}$,调节不同方法的参数使其效果达到最优。令 $n = \{1, 3, 5, 7, 10\}$,测试模型的 top- n 个性化推荐性能。模型的其余参数通过训练集进行学习,参数 ω 和 α 的取值范围为: $\omega = [0, 0.1, \dots, 1.0]$, $\alpha = [0, 0.05, \dots, 0.55]$ 。用户和项目隐特征维度 K 的取值范围为 $K = [10, 20, \dots, 100]$,近邻用户的个数 N_u 的取值范围为 $N_u = [10, 20, \dots, 100]$ 。

5.3 参数对推荐算法的影响

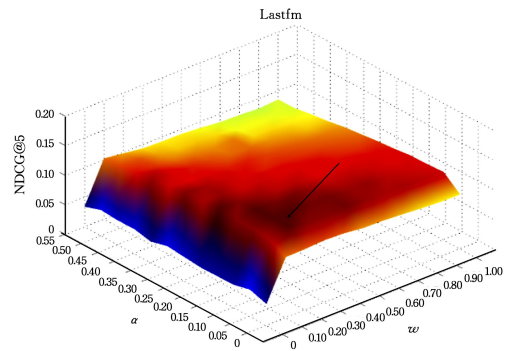
提出模型的推荐结果主要受 4 个参数的影响:1)加权矩阵分解模型的权重 ω 和初始值 α ;2)用户和项目隐特征矩阵的维度 K ;3)近邻用户的个数 N_u 。

5.3.1 ω 和 α 对推荐结果的影响

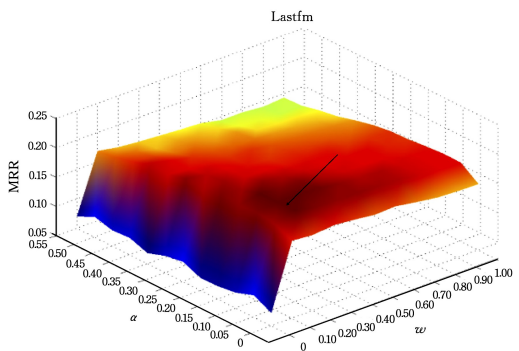
首先,分别固定用户(项目)的隐特征维度 $K = 30$ 以及近邻用户个数 $N_u = 20$,观察 WRMF-Deep 的推荐性能随 ω 和 α 变化的趋势。图 4(a)~图 4(c)分别给出在 Lastfm 数据集上($Pre@5$, $NDCG@5$ 和 MRR)推荐性能随 ω 和 α 变化的结果。箭头指向的局部坐标范围代表取得最优推荐性能的位置。



(a) $Pre@5$ 上推荐性能随 ω 和 α 变化的结果



(b) $NDCG@5$ 上推荐性能随 ω 和 α 变化的结果



(c) MRR 上推荐性能随 ω 和 α 变化的结果

图 4 ω 和 α 对推荐性能的影响

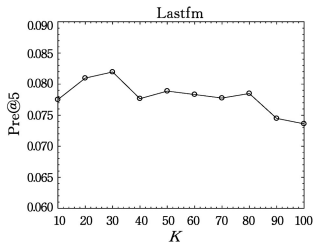
Fig. 4 Effects of parameters ω and α on recommendation performance

由图 4 可以看出,随着 w 和 α 的变化,WRMF-Deep 在 $\text{Pre}@5$, $\text{NDCG}@5$, MRR 3 个评价指标上的大致变化趋势是先增大后减小,在 $(w, \alpha) = (0.2, 0.1)$ 附近取得最优结果。具体地,固定 w , 增大 α 的值,WRMF-Deep 方法的性能先增大后减小。类似地,固定 α , 增大 w 的值,WRMF-Deep 方法性能的总体变化趋势也是先增大后减少。结果表明,参数 w 和 α 同时对 WRMF-Deep 推荐性能的提升都发挥着关键作用,另外,当 $\alpha=0$, 即同等地看待用户 u 对潜在项目 P_u^+ 和其他未观测项目 P_u^- 的偏好程度时,WRMF-Deep 的性能有所下降,表明了利用近邻关系来表达用户细粒度偏好,以及利用加权矩阵分解对用户偏好建模的有效性。

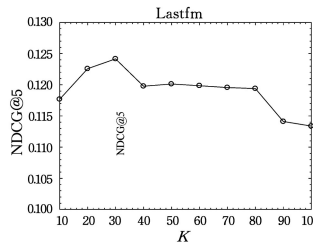
5.3.2 特征维度 K 对推荐结果的影响

确定参数 w 和 α 的值后,设定近邻用户个数 $N_u = 20$, 使隐特征维度从 10 到 100 等间距地变化,观察 WRMF-Deep 推荐性能随特征维度 K 变化的趋势。

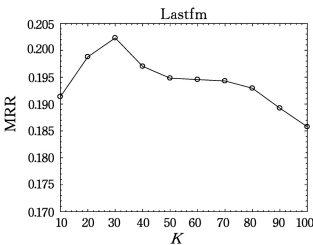
图 5(a) — 图 5(c) 分别显示了 Lastfm 数据集上 ($\text{Pre}@5$, $\text{NDCG}@5$ 和 MRR) 推荐性能随特征维度变化的结果。



(a) $\text{Pre}@5$ 上推荐性能随维度 K 变化的结果



(b) $\text{NDCG}@5$ 上推荐性能随维度 K 变化的结果



(c) MRR 上推荐性能随维度 K 变化的结果

图 5 不同特征维度 K 对推荐性能的影响

Fig. 5 Effects of dimension K on recommendation performance

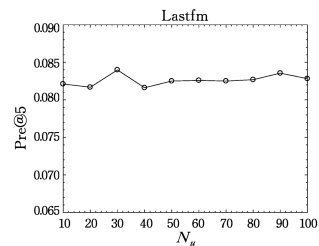
可以看到,随着隐特征维度 K 的增大,各个指标上的结果呈现先上升后下降的趋势。具体地,当维度从 10 增大到 30 时,推荐性能单调上升,说明增加模型复杂度能够增强模型刻画用户与项目交互等行为的能力;但随着 K 的增大,推荐性能总体呈下降趋势,这是因为维度的增加可能会引起过拟合的问题。另外,随着 K 的增大,算法的训练复杂度也逐渐上升。因此,将特征维度设定为 30 是一个可以接受的选择。

5.3.3 近邻用户个数 N_u 对推荐结果的影响

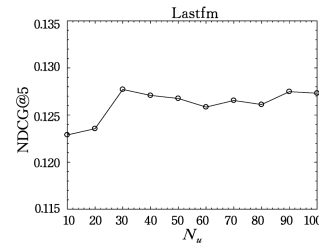
近邻用户数目直接影响着用户细粒度偏好序列的构建,进而影响着 WRMF-Deep 模型的推荐性能,其值过大或者过小都可能导致推荐的结果不理想。为确定合适的近邻用户数目,固定好参数 w, α 以及隐特征维度 $K=30$ 后,度量不同近邻数目 N_u 下推荐性能,择优选取参数。

图 6(a) — 图 6(c) 分别显示了 Lastfm 数据集上 ($\text{Pre}@5$, $\text{NDCG}@5$ 和 MRR) 推荐性能随近邻用户个数 N_u 变化的结果。

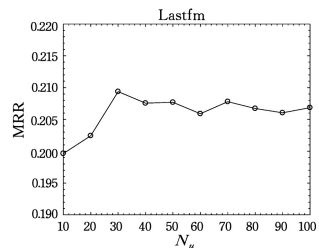
由图 6 可知,随着近邻用户个数 N_u 的增大,推荐结果整体呈现先上升后小幅波动的趋势,并在 $N_u=30$ 处取得最优效果。这是因为当近邻用户数目较小时,由近邻用户确定的用户可能感兴趣的潜在项目较少,换句话说,未观测数据未得到有效挖掘与利用,使得推荐性能不高。随着 N_u 的增大,由近邻用户确定的潜在项目增多,从未观测数据中挖掘到越来越多有用的信息,推荐性能逐渐提高。但当 N_u 进一步增大时,推荐性能反而下降,这是因为产生的噪声数据也越来越多,进而影响推荐性能。



(a) $\text{Pre}@5$ 上推荐性能随近邻用户个数 N_u 变化的结果



(b) $\text{NDCG}@5$ 上推荐性能随近邻用户个数 N_u 变化的结果



(c) MRR 上推荐性能随近邻用户个数 N_u 变化的结果

图 6 近邻用户个数 N_u 对推荐性能的影响

Fig. 6 Effects of number of neighbor users on recommendation performance

5.4 不同方法的推荐性能对比

确定好最优参数后,对比本文所提方法与其他对比方法的个性化推荐性能。图 7 和表 1 分别给出了不同指标上的对比结果。

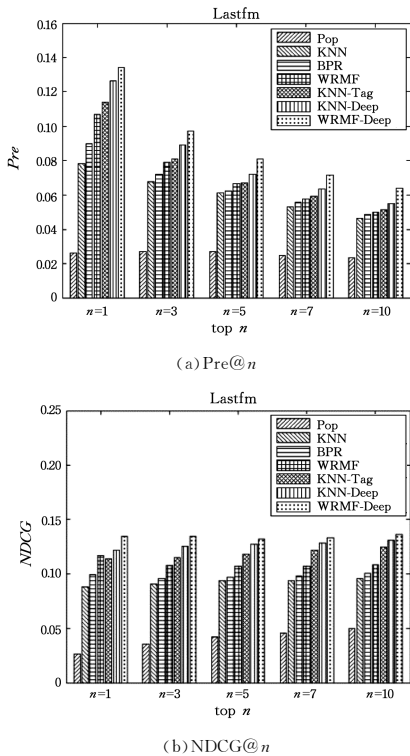


图7 不同方法的 top-n 推荐结果比较

Fig. 7 Recommendation results of different methods

表1 不同方法在 MRR 指标上推荐结果比较

Table 1 Comparison of recommendation performance on MRR

方法	MRR
Pop	0.0753
KNN	0.1606
BPR	0.1704
WRMF	0.1876
KNN-Tag	0.1849
KNN-Deep	0.1908
WRMF-Deep	0.2071

由图7和表1可以看出:

1)在多个评价指标上,WRMF-Deep方法的性能均优于其他对比方法,证明了本文方法的合理性和有效性。一方面,WRMF-Deep方法能充分利用加权矩阵分解和 k -近邻方法的优势对用户的细粒度偏好进行建模,缓解用户-项目交互数据的稀疏不平衡问题的影响;另一方面,WRMF-Deep能利用深度学习的方法缓解标签数据的冗余、语义模糊等问题。

2)WRMF-Deep方法优于KNN-Deep。这两种方法均采用深度学习的方法来解决标签数据的冗余、语义模糊等问题。不同的是,本文提出的方法WRMF-Deep在处理稀疏不平衡数据方面具有一定的优势,能够利用近邻用户关系从大规模未观测数据中挖掘用户可能感兴趣的项目,更加准确地对用户细粒度偏好进行建模,缓解数据不平衡性对用户偏好建模的不利影响。

3)KNN-Deep方法的推荐性能优于KNN-Tag方法,证明了利用深度学习技术能够很好地解决标签数据冗余、语义模糊等问题,提高用户相似度计算的准确性,进而提高推荐的准确性。同时,为本文方法利用近邻用户关系获取更为准确

的用户细粒度偏好提供技术支撑。

4)KNN-Tag方法的推荐性能优于KNN方法,这是因为前者利用标签数据辅助隐式反馈数据进行推荐,后者仅利用隐式反馈数据进行推荐,说明了合理利用辅助数据可有效地改善推荐性能。

4)相对于传统的协同过滤推荐技术(包括KNN,BPR),WRMF的表现相对更优,这在一定程度上说明,针对本文采用的稀疏不平衡数据集,WRMF的适应性更强,能更准确地刻画用户与项目的交互行为。

5)所有的对比方法中,Pop方法的推荐性能最差,这是因为Pop是一种非个性化推荐方法,其推荐结果与每个用户的个性化行为与需求无关,因此推荐结果通常无法很好地满足用户需求。而其他方法均根据每个用户的个性化行为并协同其他用户行为进行推荐。

结束语 面向隐式反馈的个性化推荐是当前推荐领域的研究热点之一,利用标签等显式反馈数据来辅助推荐,有利于提高推荐性能。为了缓解隐式反馈数据的稀疏不平衡,以及标签数据的冗余、语义模糊等的影响,提出了一种基于用户细粒度偏好和加权矩阵分解的个性化推荐方法。利用近邻用户关系挖掘用户对项目的细粒度偏好关系,缓解了隐式反馈数据稀疏不平衡的影响,实现对用户偏好更加准确的建模;利用深度学习的方法抽取用户-标签的深层语义特征,缓解了原始标签数据的冗余、语义模糊对用户表征的影响,提高了近邻用户获取的准确性;最后,提出增量加权矩阵分解模型对细粒度偏好关系进行建模与推荐。实验结果表明,所提方法有效缓解了上述问题的影响,在多个推荐评价指标上的结果优于对比方法。但所提算法并不区分不同近邻用户的重要程度,在未来工作中,将考虑对标注行为相近的用户行为赋予更大的置信权重,以进一步细化用户偏好关系,进而改善细粒度加权矩阵分解模型的推荐准确性。

参考文献

- [1] HU Y,KOREN Y,VOLINSKY C.Collaborative filtering for implicit feedback datasets[C]// Proceedings of the 8th International Conference on Data Mining, 2008.
- [2] PAN R,ZHOU Y,CAO B,et al.One-Class Collaborative Filtering[C]// International Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, 2008.
- [3] 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.
- [4] PAN W,ZHONG H,XU C,et al.Adaptive Bayesian personalized ranking for heterogeneous implicit feedbacks[J]. Knowledge-Based Systems, 2015,73(1):173-180.
- [5] SHEPITSEN A,GEMMELL J,MOBASHER B,et al.Personalized recommendation in social tagging systems using hierarchical clustering[C]//Proceeding of the 2008 ACM Conference on

- Recommender Systems, 2008;259-266.
- [6] ZUO Y,ZENG J,GONG M,et al. Tag-aware recommender systems based on deep neural networks[J]. *Neurocomputing*,2016, 204:51-60.
- [7] ZHANG Z K,ZHOU T,ZHANG Y C. Personalized recommendation via integrated diffusion on user-item-tag tripartite graphs [J]. *Physica A:Statistical Mechanics and its Applications*,2010, 389(1):179-186.
- [8] ZHANG Z,ZENG D D,ABBASI A,et al. A Random Walk Model for Item Recommendation in Social Tagging Systems[J]. *Acm Transactions on Management Information Systems*,2013, 4(2):1-24.
- [9] HE M,YANG F,YAO K S,et al. TEFRCF:Collaborative filtering personalized recommendation algorithm based on tag entropy feature representation[J]. *Computer Science*,2018,45(6): 465-486. (in Chinese)
何明,杨芃,要凯升,等. TEFRCF:标签熵特征表示的协同过滤个性化推荐算法[J]. *计算机科学*,2018,45(6):465-486.
- [10] ZENG D,LI H. How Useful Are Tags? — An Empirical Analysis of Collaborative Tagging for Web Page Recommendation[C]// *IEEE Isi 2008 Paisi, Pacif, and Soco International Workshops on Intelligence and Security Informatics*, 2008;320-330.
- [11] HE M,YAO K S,YANG F,et al. Collaborative filtering personalized recommendation based on similarity of tag information feature[J]. *Computer Science*,2018,45(6):415-422. (in Chinese)
何明,要凯升,杨芃,等. 基于标签信息特征相似性的协同过滤个性化推荐[J]. *计算机科学*,2018,45(6):415-422.
- [12] WU L,CHEN E,LIU Q,et al. Leveraging tagging for neighborhood-aware probabilistic matrix factorization[C]// *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, 2012;1854-1858.
- [13] MA H,ZHOU T C,LYU M R,et al. Improving Recommender Systems by Incorporating Social Contextual Information[J]. *ACM Transactions on Information Systems*,2011,29(2):1-23.
- [14] ZHANG Z,ZENG D D,ABBASI A,et al. A random walk model for item recommendation in social tagging systems[J]. *ACM Transactions on Management Information Systems*,2013,4(2): 1-24.
- [15] LI H Y,FU Y Q. Collaborative filtering recommendation algorithm based on tag clustering and item topic[J]. *Computer Science*,2018,45(4):247-251. (in Chinese)
- 李昊阳,符云清. 基于标签权重评分的推荐模型及算法研究[J]. *计算机科学*,2018,45(4):247-251.
- [16] ELKAHKY A M,SONG Y,HE X. A multi-view deep learning approach for cross domain user modeling in recommendation systems[C]// *WWW'15*. 2015;278-288.
- [17] HUANG P S,HE X,GAO J,et al. Learning deep structured semantic models for web search using clickthrough data [C] // *ACM International Conference on Conference on Information & Knowledge Management*. 2013;2333-2338.
- [18] HUANG L W,JIANG B T,LV S Y,et al. Survey on deep learning based recommender systems[J]. *Chinese Journal of Computers*,2018,427(7):191-219. (in Chinese)
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*,2018,427(7):191-219.
- [19] LE Q V,NGIAM J,COATES A,et al. On optimization methods for deep learning [C] // *International Conference on Machine Learning(ICML 2011)*. Bellevue, Washington, USA, 2011;265-272.
- [20] HINTON G E. A Practical Guide to Training Restricted Boltzmann Machines[J]. *Momentum*,2012,9(1):599-619.
- [21] SEDHAIN S,MENON A K,SANNER S,et al. On the effectiveness of linear models for one-class collaborative filtering[C]// *Proceedings of the 30th Conference on Artificial Intelligence*. 2016.
- [22] LI H,GE Y,HONG R,et al. Point-of-Interest Recommendations:Learning Potential Checkins from Friends[C]// *Proceedings of the 22nd International ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2016.
- [23] DEVOOGHT R,KOURTELLIS N,MANTRACH A. Dynamic matrix factorization with priors on unknown values[C]// *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM,2015:189-198.
- [24] DEVOOGHT R,KOURTELLIS N,MANTRACH A. Dynamic Matrix Factorization with Priors on Unknown Values [C] // *Proceedings of the 21st International ACM SIGKDD Conference on Knowledge Discovery and Data Mining Sydney, NSW, Australia*,2015.
- [25] RENDLE S,FREUDENTHALER C,GANTNER Z,et al. BPR: Bayesian Personalized Ranking from Implicit Feedback [C] // *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence*. 2009;452-461.