

面向协同过滤推荐的新型混合评分函数

肖诗涛¹ 邵莹侠¹ 宋卫平² 崔 斌²

1 北京邮电大学计算机学院 北京 100876

2 北京大学信息科学技术学院 北京 100871

(stxiao@bupt.edu.cn)

摘要 协同过滤技术在现代推荐系统中得到了广泛的应用,其基本思想是相似的用户会喜欢相似的物品。评分函数(Score Function, SF)是协同过滤推荐模型的一个关键技术,用于评估用户对物品的喜好程度。然而,目前常用的评分函数存在如下缺陷,即内积评分函数难以有效捕捉用户与用户以及物品与物品的相似度,而欧几里德距离度量函数由于几何空间限制降低了模型的表达能力。文中提出了一种融合内积相似度和欧几里德距离度量的新颖的混合评分函数,并从理论上分析了此混合评分函数的性质,证明它能有效弥补现有评分函数的不足。此外,新的混合评分函数是一项通用技术,适用于诸多现有的推荐模型(如 SVD++, MF, NGCF, CML 等),能够提高模型的推荐质量。最后,在 6 个公开数据集上进行了大量实验,验证了新混合评分函数的优越性能。

关键词: 推荐系统; 协同过滤; 评分函数

中图法分类号 TP391.3

Hybrid Score Function for Collaborative Filtering Recommendation

XIAO Shi-tao¹, SHAO Ying-xia¹, SONG Wei-ping² and CUI Bin²

1 School of Computer Science, Beijing University of Posts and Telecommunications, Beijing 100876, China

2 School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China

Abstract Collaborative Filtering has been widely used in modern recommendation systems, and it assumes that similar users prefer similar items. A key ingredient of CF-based recommendation model is the score function, which measures the preference of users on items. However, there are some shortages in the most popular score functions. The inner product score function fails to capture the user-user similarity and item-item similarity effectively, and Euclidean distance measurement function reduces the expressiveness of the model because of its geometrical restriction. This paper proposes a novel hybrid score function by mixing the inner product-based similarity and the Euclidean distance metric, and further theoretically analyze its properties, thus proving that the new score function can avoid the aforementioned shortages effectively. In addition, the new hybrid score function is a general technique and can help to improve the quality of recommendation for existing models (e.g., SVD++, MF, NGCF, CML). Extensive empirical studies over 6 datasets demonstrate the superior performance of the proposed hybrid score function.

Keywords Recommendation system, Collaborative filtering, Score function

1 引言

推荐系统作为解决信息爆炸问题的一种标准解决方案,已被广泛应用于电子商务、广告、社交媒体等诸多在线服务^[1]。它通过用户的历史交互数据(如点击、浏览、购买等)发现用户和物品间的联系。协同过滤(Collaborative Filtering, CF)是一种成功的推荐技术,通过假设相似的用户会喜欢相似的物品来实现推荐^[2]。在众多 CF 实现方法中,矩阵分解(Matrix Factorization, MF)是最流行的方法之一^[3]。MF 将

每个用户和物品投射到一个低维向量空间中,并使用一个评分函数来捕获用户和物品之间的交互关系。

评分函数的选择会极大地影响物品的排名,并最终影响推荐的质量。内积是一个标准的评分函数,已被应用于很多推荐模型中^[4-6]。然而,这种简单的评分函数限制了推荐模型的性能^[7]。内积难以有效地捕捉用户与用户之间以及物品与物品之间的相似度,从而导致了推荐精度的损失^[8]。另一种常见的评分函数是使用度量学习(Metric Learning, ML)中的欧几里德距离,此评分函数通过最小化用户与物品间的欧氏

到稿日期:2020-09-08 返修日期:2020-10-03 本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。

基金项目:国家自然科学基金(U1936104,61702015);中央高校基本科研业务费专项资金资助(2020RC25)

This work was supported by the National Natural Science Foundation of China(U1936104,61702015) and Fundamental Research Funds for the Central Universities(2020RC25).

通信作者:邵莹侠(shaoyx@bupt.edu.cn)

距离,将所有相似的用户或物品聚集在一起,从而使模型能够学习向量空间中用户与用户之间以及物品与物品之间的相似性^[8]。然而,如果直接优化欧氏距离,由于用户和物品之间的最优距离是零,模型会将每对交互中的用户和物品优化到一个点上。这种几何空间限制(Geometrical Restriction,GR)^[9]制约了模型的表达能力,也导致了推荐模型的准确度降低。

为了解决上述问题,本文提出了一种融合内积相似性和欧氏距离度量的混合评分函数。此新颖的混合评分函数能有效地捕获用户与用户之间以及物品与物品之间的相似性,同时减弱了欧氏距离度量的几何空间限制。本文对新提出的混合评分函数进行了详实的理论分析,并介绍了它的两个重要特性:1)混合评分函数保证了用户与用户之间以及物品与物品之间相似性的下界;2)混合评分函数保证一对交互记录不会被优化到几何空间中的同一点,有效减弱了欧氏距离的几何限制。混合评分函数是首个融合距离度量和内积相似性的评分函数。为了验证新评分函数的有效性和通用性,本文将混合评分函数应用于4个推荐模型(SVD++,MF,NGCF,CML),并在6个公共数据集上进行了综合实验。实验结果表明,新的评分函数显著提高了推荐结果的质量。

2 背景与相关工作介绍

2.1 基于隐式反馈的推荐

隐式反馈的交互数据包括购买历史、浏览历史等,间接地反映了用户的偏好和物品的属性特征。本文使用字母 u 和 i 分别代表用户和物品,用户集和物品集分别用 U 和 I 表示。隐式反馈矩阵 \mathbf{Y} 是一个 $|U| \times |I|$ 大小的矩阵,其各个元素的值为:

$$y_{ui} = \begin{cases} 1, & \text{如果 } u \text{ 和 } i \text{ 之间有过交互记录} \\ 0, & \text{如果 } u \text{ 和 } i \text{ 之间没有交互记录} \end{cases} \quad (1)$$

其中, y_{ui} 的值为0,表示用户 u 和物品 i 之间没有产生过交互,但这并不意味着 u 不喜欢 i ,在大多数情况下,用户只是不知道该物品的存在。由此,引出了基于隐式反馈的推荐任务:估计矩阵 \mathbf{Y} 中值为0的元素的得分,预测产生交互的概率。

换言之,给定隐式反馈矩阵 \mathbf{Y} ,推荐模型的目标是学习一个评分函数(或预测函数): $\hat{y}_{ui} = S(\mathbf{p}_u, \mathbf{q}_i, \Theta)$,其中 \mathbf{p}_u 和 \mathbf{q}_i 分别表示用户 u 和物品 i 的嵌入向量, \hat{y}_{ui} 是用户 u 对物品 i 的喜好程度, Θ 是模型参数。其中,最常用的评分函数有内积函数和欧氏距离函数。

1) 内积评分函数

内积是MF中的一个标准评分函数,它通过向量内积来估计分数,具体公式如下:

$$S_m(\mathbf{p}_u, \mathbf{q}_i) = \mathbf{p}_u^\top \mathbf{q}_i \quad (2)$$

2) 欧氏距离评分函数

欧氏距离 $d_{ui}^2 = \|\mathbf{p}_u - \mathbf{q}_i\|^2$ 是另一种常见的评分方法。其目标是减小交互记录中的用户和物品间的距离,同时扩大没有交互关系的用户和物品间的距离。由于越小的距离代表着发生交互的可能性越高,欧氏距离评分函数可描述为:

$$S_{ed}(\mathbf{p}_u, \mathbf{q}_i) = -d_{ui}^2 \quad (3)$$

2.2 相关工作

本节介绍了使用不同评分函数的协同过滤推荐模型。基

于潜在因子模型的CF通常采用内积作为评分函数。MF^[5]是一个著名的潜在因子模型,它先将用户和物品映射到一个潜在因子空间,然后利用内积函数来估计分数。近年来,内积也逐渐被用于神经网络推荐模型。例如,NGCF(Neural Graph Collaborative Filtering)^[10]利用图结构信息来获得更好的用户和物品表示,并应用内积来估计用户对目标物品的偏好。CDL(Collaborative Deep Learning)^[11]利用深度学习技术来学习内容信息,同样使用内积来预测分数。

除内积外,采用欧氏距离作为评分函数的方法也受到了很多关注。Khoshneshin等使用欧氏距离学习嵌入向量,并应用于显式反馈数据上^[12]。Cen等使用欧氏距离计算用户最有可能点击的物品^[13]。在隐式反馈的协同过滤问题中,CML(Collaborative Metric Learning)是当前最先进的度量学习模型,它将用户和物品编码在一个联合空间内,并优化产生过交互关系的用户和物品之间的欧氏距离^[8]。

此外,随着神经网络技术的广泛应用,最近也出现了基于神经网络的评分函数。例如,NMF(Neural Matrix Factorization)^[7]将内积替换为一个非线性的神经网络函数。

3 经典评分函数的缺陷分析

3.1 内积评分函数

虽然内积评分函数已在推荐模型中得到广泛应用^[4-6],但它不能有效地捕获用户与用户之间以及物品与物品之间的相似度^[8]。下面通过一个简化的例子进行分析,具体内容如图1所示。

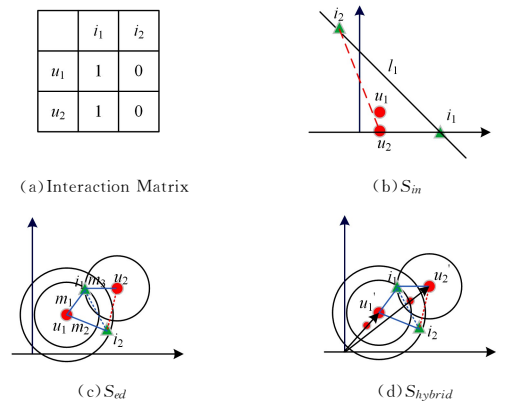


图1 在二维向量空间中不同评分函数的3个交互的比较
Fig.1 Comparison of different score functions in two-dimensional vector space with three interactions

例1 给定3个交互 $(u_1, i_1), (u_1, i_2), (u_2, i_1)$,并使用二维向量表示用户和物品。在模型训练之后,在理想情况下,3个观察到的交互的内积都是1,例如, $S_m(\mathbf{p}_{u_1}, \mathbf{q}_{i_2}) = 1$ 。图1(b)可视化了一组可能的用户和物品的潜在向量,即 $\mathbf{p}_{u_1} = [0.5, 0.5], \mathbf{p}_{u_2} = [0.5, 0], \mathbf{q}_{i_1} = [2, 0], \mathbf{q}_{i_2} = [-0.5, 2.5]$ 。

因为 u_1 既喜欢 i_1 又喜欢 i_2 ,所以可以认为这两个物品在某种程度上是相似的。根据协同过滤原理,由于 u_2 也喜欢 i_1 ,则给 u_2 推荐相似的物品 i_2 。但是,从上面的例子可以发现 $S_m(\mathbf{q}_{i_1}, \mathbf{q}_{i_2}) = -1$,而 $S_m(\mathbf{q}_{i_1}, \mathbf{q}_{i_2}) = -1$ 意味着 i_1 和 i_2 并不相似。同时, $S_m(\mathbf{p}_{u_2}, \mathbf{q}_{i_2}) = -0.25$,这意味着 u_2 不喜欢物品 i_2 。更糟糕的是,如果 i_2 沿着直线 $x + y = 2$ 移动,尽管它

仍然满足约束 $S_m(\mathbf{p}_{u_1}, \mathbf{q}_{i_2})=1$, 但 $S_m(\mathbf{q}_{i_1}, \mathbf{q}_{i_2})$ 和 $S_m(\mathbf{p}_{u_2}, \mathbf{q}_{i_2})$ 都将不受约束, 即 $S_m(\mathbf{q}_{i_1}, \mathbf{q}_{i_2}) \in [-\infty, +\infty]$, 并且 $S_m(\mathbf{p}_{u_2}, \mathbf{q}_{i_2}) \in [-\infty, +\infty]$. 因此, 内积作为评分函数, 很难捕捉到物品与物品之间以及用户与用户之间的相似性, 从而导致模型精度下降。

3.2 欧氏距离评分函数

为了弥补内积评分函数的不足, 部分研究通过度量学习引入了欧氏距离评分函数。因为欧氏距离满足三角不等式, 即 $d_{u_1} + d_{u_2} \geq d_{i_1, i_2}$, 当用户 u 既喜欢 i_1 又喜欢 i_2 时, 欧氏距离评分函数能够确保 i_1 和 i_2 之间的相似性有一个下界。换句话说, i_1 在某种程度上类似于 i_2 。因此, 它能比内积更有效地捕获数据之间的潜在关系。在此, 使用例 2 进行分析。

例 2 同样使用例 1 中的 3 个交互数据, 并假设 $S_{ed}(\mathbf{p}_{u_1}, \mathbf{q}_{i_1})=M_1, S_{ed}(\mathbf{p}_{u_1}, \mathbf{q}_{i_2})=M_2, S_{ed}(\mathbf{p}_{u_2}, \mathbf{q}_{i_1})=M_3$ 。根据三角不等式, 我们可以得到:

$$S_{ed}(\mathbf{q}_{i_1}, \mathbf{p}_{i_2}) \geq M_1 + M_2 - 2\sqrt{M_1 M_2} = f(M_1, M_2)$$

即 i_1 与 i_2 的相似度存在一个下界。在此基础上, 我们可以进一步得到:

$$S_{ed}(\mathbf{p}_{u_1}, \mathbf{q}_{i_2}) \geq f(f(M_1, M_2), M_3)$$

即 u_2 与 i_2 的分数同样存在一个下界, 这意味着由于 i_2 类似于 u_2 曾经交互过的 i_1 , 因此 u_2 会倾向于喜好 i_2 。

虽然通过距离评分函数可以学习用户与用户之间以及物品与物品之间的相似性, 但是它仍会限制模型的表达能力。欧氏距离函数试图将每对交互关系中的用户和物品拟合到向量空间中的同一点^[9]。例如, 当我们优化距离函数时, (u_1, i_1) 和 (u_1, i_2) 都将倾向于投射到同一个点上, 这种情况能使得距离最小, 这样 u_1, i_1 和 i_2 最终都将优化成同一个点。因此, 欧氏距离评分函数对于交互关系的约束过于严格, 以至于难以实现很好的数据拟合。

4 基于混合评分函数的推荐

4.1 混合评分函数

如前文所述, 内积评分函数无法捕捉到数据间的相似性, 而欧氏距离评分函数会将用户与物品拟合到同一点, 它们都限制了模型的表达能力。因此, 需要一个更灵活、更具表现力的评分函数来建模用户和物品之间的关系。本文提出了一个简单且有效的混合评分函数。

$$S_{hybrid}(\mathbf{p}_u, \mathbf{q}_i) = (1-\gamma)S_m(\mathbf{p}_u, \mathbf{q}_i) + \gamma S_{ed}(\mathbf{p}_u, \mathbf{q}_i) \quad (4)$$

其中, γ 是权重参数, 范围为 $[0, 1]$; S_m 和 S_{ed} 分别是内积评分函数和欧氏距离评分函数。注意, 当 $\gamma=1$ 时, S_{hybrid} 退化为欧氏距离评分函数, 而当 $\gamma=0$ 时, 它是内积评分函数。

接下来, 本文进一步从理论上分析了新混合评分函数的性质。

1) 相似度有下界

假设 (u, i) 在混合评分函数下的得分为 M , 那么评分函数可改写为:

$$S_{hybrid}(\mathbf{p}_u, \mathbf{q}_i) = M$$

$$(1-\gamma)\mathbf{p}_u^\top \mathbf{q}_i - \gamma \|\mathbf{p}_u - \mathbf{q}_i\|^2 = M$$

$$-\gamma \mathbf{p}_u^\top \mathbf{p}_u + (1+\gamma)\mathbf{p}_u^\top \mathbf{q}_i - \gamma \mathbf{q}_i^\top \mathbf{q}_i = M$$

$$\sum_{k=1}^K (q_{ik} - \frac{1+\gamma}{2\gamma} p_{uk})^2 = (\frac{(1+\gamma)^2}{4\gamma^2} - 1) \sum_{k=1}^K (p_{uk})^2 - \frac{M}{\gamma}$$

其中, K 是用户和物品潜在向量的维数。这里定义一个函数:

$$r(M, \gamma, \mathbf{p}_u) = \sqrt{(\frac{(1+\gamma)^2}{4\gamma^2} - 1) \sum_{k=1}^K (p_{uk})^2 - \frac{M}{\gamma}}$$

因此, 可以得到:

$$\sum_{k=1}^K (q_{ik} - \frac{1+\gamma}{2\gamma} p_{uk})^2 = r(M, \gamma, \mathbf{p}_u)^2 \quad (5)$$

根据式(5), 函数 r 可以看作是 \mathbf{q}_i 和 $\frac{1+\gamma}{2\gamma}\mathbf{p}_u$ 在欧氏空间中的距离。如图 1(d) 所示, 给定用户向量 \mathbf{p}_u 和其与物品 i 的分数 M , 那么物品 i 将会位于空间内的一个圆上, 圆的中心是 $u' = \frac{1+\gamma}{2\gamma}\mathbf{p}_u$, 半径是 $r(M, \gamma, \mathbf{p}_u)$ 。

回顾第 3.2 节中的例 2, 欧氏距离由于满足三角不等式, 可以很好地捕捉到数据 i_1 和 i_2 的相似性。在混合评分函数中, 我们同样可以证明当用户 u_1 与物品 i_1 和 i_2 都有过交互关系时, i_1 和 i_2 的相似度存在下界, 具体公式如下:

$$S_{hybrid}(\mathbf{q}_{i_1}, \mathbf{q}_{i_2}) \geq g(M_1, M_2, \Theta) \quad (6)$$

其中,

$$g(M_1, M_2, \Theta) = \frac{-1-\gamma}{2}(r(M_1, \gamma, \mathbf{p}_{u_1}) + r(M_2, \gamma, \mathbf{p}_{u_1}))^2$$

证明: 首先, 根据式(6), 因为 i_1 和 i_2 都被用户 u_1 交互过, 它们在向量空间内与点 $u_1' = \frac{1+\gamma}{2\gamma}\mathbf{p}_{u_1}$ 的距离分别为 $r(M_1, \gamma, \mathbf{p}_{u_1})$ 和 $r(M_2, \gamma, \mathbf{p}_{u_1})$, 因此这两个物品在向量空间的距离不会超过 $r(M_1, \gamma, \mathbf{p}_{u_1}) + r(M_2, \gamma, \mathbf{p}_{u_1})$ 。基于此, 可以得到:

$$\begin{aligned} S_{hybrid}(\mathbf{q}_{i_1}, \mathbf{q}_{i_2}) &= (1-\gamma)\mathbf{q}_{i_1}^\top \mathbf{q}_{i_2} - \gamma \|\mathbf{q}_{i_1} - \mathbf{q}_{i_2}\|^2 \\ &\geq -\frac{1-\gamma}{2} \|\mathbf{q}_{i_1} - \mathbf{q}_{i_2}\|^2 - \gamma \|\mathbf{q}_{i_1} - \mathbf{q}_{i_2}\|^2 \\ &= \frac{-1-\gamma}{2} \|\mathbf{q}_{i_1} - \mathbf{q}_{i_2}\|^2 \\ &\geq \frac{-1-\gamma}{2} (r(M_1, \gamma, \mathbf{p}_{u_1}) + r(M_2, \gamma, \mathbf{p}_{u_1}))^2 \end{aligned}$$

通过重复嵌套这个三角不等式, 可以得到 (u_2, i_2) 的分数的下界, 即:

$$S_{hybrid}(\mathbf{p}_{u_2}, \mathbf{q}_{i_2}) \geq g(g(M_1, M_2, \Theta), M_3, \Theta)$$

与内积相比, 混合评分函数可以学习 i_1 和 i_2 之间的相似性, 生成更符合协同过滤原理的推荐结果, 即向 u_2 推荐 i_2 。与距离评分函数相比, 混合评分函数的下界可以通过参数 γ 进行调整, 其设置更加灵活。

2) 有交互关系的用户和物品不会被拟合到向量空间的同一点

将混合评分函数(5)展开为:

$$\begin{aligned} S_{hybrid}(\mathbf{p}_u, \mathbf{q}_i) &= (1-\gamma)\mathbf{p}_u^\top \mathbf{q}_i - \gamma \|\mathbf{p}_u - \mathbf{q}_i\|^2 \\ &= -\gamma \left\| \mathbf{q}_i - \frac{1+\gamma}{2\gamma} \mathbf{p}_u \right\|^2 + \frac{4\gamma^2 - (1+\gamma)^2}{4\gamma} \|\mathbf{p}_u\|^2 \end{aligned} \quad (7)$$

当优化 $S_{hybrid}(\mathbf{p}_u, \mathbf{q}_i)$ 时, 为获得更大的分数, \mathbf{q}_i 将会趋向于 $\frac{1+\gamma}{2\gamma}\mathbf{p}_u$ 。与距离评分函数相比, γ 相当于一个放缩系数。

由于它的存在, \mathbf{q}_i 不会被直接优化成 \mathbf{p}_u 。注意, 当 $\gamma=1$ 时, 混合评分函数退化成距离评分函数。

4.2 MixRec: 基于混合评分函数的推荐模型

本文提出的混合评分函数与具体的推荐模型是正交的,

它可以应用于许多已有的模型,以提高推荐的质量。根据等式(5),可将混合评分函数应用于任何显式的学习用户和物品表示(即 p_u 和 q_i)的推荐模型,如 MF, SVD++, NGCF 和 CML 等。本文使用 MixRec(A)表示一个使用混合评分函数的模型 A,如 MixRec(MF), MixRec(CML)。

在没有明确说明的情况下,本文将隐式反馈的推荐模型视为一个二元分类问题,并使用交叉熵作为损失函数。利用 sigmoid 函数将分数缩放到 $[0, 1]$ 的范围内: $y'_{ui} =$

$\frac{1}{1 + e^{-S_{\text{Hybrid}}(p_u, q_i)}}$ 。具体的损失函数如下:

$$L = - \sum_{(u,i) \in O^+ \cup O^-} [y_{ui} \log(y'_{ui}) + (1 - y_{ui}) \log(1 - y'_{ui})] + \lambda \|\Theta\|^2$$

其中, O^+ 表示观察到的交互集合, O^- 表示负面实例的集合, 可以从未观察到的交互中采样, Θ 是模型参数, λ 控制正则化强度以防止过拟合。值得注意的是, 使用交叉熵作为损失函数并不是必须的, 其他损失函数如贝叶斯个性化排序^[5]和铰链损失^[8]也都可以使用。

5 实验结果

5.1 实验设置

5.1.1 数据集

为了评估混合评分函数的有效性, 本文在 6 个公开数据集上进行了实验: CiteULike-a^[11], Epinion^[14], LastFM, BookCX^[15], Amazon video 和 Sports^[16]。表 1 列出了数据集的统计信息。

表 1 数据集信息

Table 1 Statistics of datasets

数据集	交互数量	用户数量	物品数量	稀疏度/%
Citeulike-a	204 987	5 551	16 980	99.783
Epinions	664 823	40 163	139 738	99.988
Video	21 809	1 372	7 957	99.800
Sports	313 131	19 170	109 739	99.985
LastFM	92 834	1 892	17 632	99.722
BookCX	881 843	11 884	291 386	99.975

表 2 性能对比

Table 2 Performance comparison

	Video		Sports		LastFM		CiteULike-a		Epinions		BookCX	
	HR@5	NDCG@5	HR@5	NDCG@5	HR@5	NDCG@5	HR@5	NDCG@5	HR@5	NDCG@5	HR@5	NDCG@5
UserKNN	0.4585	0.3666	0.2168	0.1857	0.7160	0.6202	0.7244	0.6230	0.4015	0.3725	0.3225	0.2855
ItemKNN	0.4745	0.3833	0.2562	0.2156	0.7314	0.6261	0.7485	0.6233	0.4854	0.4354	0.3898	0.3249
MF	0.5029	0.3880	0.4488	0.3557	0.7462	0.6336	0.8088	0.6730	<u>0.6348</u>	<u>0.5133</u>	0.5393	0.4318
SVD++	<u>0.5306</u>	<u>0.4062</u>	0.4482	0.3452	0.7531	0.6275	0.7843	0.6140	0.6145	0.4751	0.5125	0.3863
NGCF	0.4978	0.3875	<u>0.4582</u>	<u>0.3636</u>	0.7468	0.6276	0.7833	0.6259	0.6116	0.4918	0.5170	0.4106
CMN	0.4913	0.3811	0.3426	0.2626	0.7458	0.6280	0.8090	0.6589	0.5770	0.4552	0.4852	0.3923
NeuMF	0.4206	0.3099	0.3666	0.2730	0.7017	0.5839	0.7849	0.6496	0.5846	0.4577	0.5089	0.3998
CML	0.4869	0.386	0.3968	0.3122	<u>0.7574</u>	<u>0.6468</u>	<u>0.8214</u>	<u>0.6758</u>	0.5599	0.4548	<u>0.5399</u>	<u>0.4387</u>
MixRec(MF)	0.5539	0.4458	0.4899	0.3853	0.7791	0.6614	0.8310	0.6856	0.6532	0.5222	0.5875	0.4640
MixRec(SVD++)	0.5525	0.4405	0.4840	0.3842	0.7781	0.6614	0.8314	0.6931	0.6608	0.5359	0.5664	0.4473
MixRec(NGCF)	0.5292	0.4153	0.4547	0.3546	0.7510	0.6431	0.7842	0.6334	0.6151	0.4961	0.5364	0.4337
MixRec(CML)	0.5430	0.4292	0.4718	0.3697	0.7707	0.6551	0.8238	0.6754	0.6200	0.4923	0.5661	0.4496
Δ (MF)	+10.14	+14.90	+9.16	+8.32	+4.41	+4.39	+2.74	+1.87	+2.90	+1.73	+8.94	+7.46
Δ (SVD++)	+4.13	+8.44	+7.99	+11.30	+3.32	+5.40	+6.01	+12.88	+7.53	+12.80	+10.52	+15.79
Δ (NGCF)	+6.31	+7.17	-0.76	-2.48	+0.56	+2.47	+0.11	+1.20	+0.57	+0.87	+3.75	+5.63
Δ (CML)	+11.52	+11.05	+18.90	+18.42	+1.76	+1.28	+0.29	-0.06	+10.73	+8.25	+4.85	+2.48

首先,从 $\Delta(A)$ 的结果可以看出,大多数情况下,混合评分函数有助于提高推荐模型的准确性。例如,在数据集 Video 上, MixRec(MF) 与 MF 相比, HR 和 NDCG 值分别提高了

5.1.2 评测指标

借鉴相关工作^[2,7],对于每个用户,本文随机选取一个与其交互过的物品组成测试集,将剩下的交互过的物品组成训练集。本文对每个用户随机采样 100 个未交互过的物品,并将它们与测试集中的物品一起排序^[7]。在训练过程中,为每个用户随机抽取一个交互物品作为验证数据集^[2],并选择在验证集上表现得最好的模型进行测试。

本文采用 Hit Ratio (HR) 和 Normalized Discounted Cumulative Gain (NDCG)^[17] 来评估 Top-K 推荐的性能。实验结果中的数据是测试集中所有用户的平均分数。

5.1.3 基准模型和参数设置

为了验证新混合评分函数的性能,选择了 8 个协同过滤模型作为基准进行对比,它们分别是: UserKNN^[18], ItemKNN^[19], MF^[5], SVD++^[20], CML^[8], NeuMF^[7], NGCF^[10] 和 CMN^[2]。如第 4.2 节的分析,其中 4 个模型可使用本文的新评分函数,它们分别是 MF, SVD++, CML 和 NGCF。

本文将用户和物品的表示向量维度设置为 50,并对所有模型参数进行了网格搜索:学习率和正则化参数分别从 $[0.1, 0.01, 0.001, 0.0001]$ 和 $[0.1, 0.01, 0.001, 0.0001]$ 中进行选择。对于混合评分函数,从 0.1 逐步调整到 0.9。本文将 NGCF 的深度设置为 3。对于 NeuMF, 本文使用原文建议的具有金字塔结构的 3 层全连接^[7]。CML 从 $[0.1, 0.2, 0.5, 1]$ 中选择边距大小。对于 UserKNN 和 ItemKNN, 邻域大小范围为 5~100。由于篇幅限制,具体的参数调优的实验结果未在文中展示。

5.2 模型比较

本文在 6 个数据集上测量了 Top-5 推荐的性能, HR@5 和 NDCG@5 的结果如表 2 所列。其中,最好的结果用粗体突出显示,对比模型中最好的结果用下划线显示。最后 4 行为使用混合评分函数后与原模型 A 相比所得到的性能提升(%)。事实上, HR@10 和 NDCG@10 有相似的趋势,但由于篇幅有限,本文并没有列出实验结果。

10.14% 和 14.90%。性能提升的主要原因是新的评分函数可以更有效地捕捉物品和用户之间的相似性,这点将在第 5.3 节进行深入分析。其次, NGCF 使用混合评分函数后在

某些数据集上(如 Sports, LastFM)的性能提升并不显著。一个可能的原因是 NGCF 中的图神经网络架构对冗余信息进行了编码,从而影响了混合评分函数的相似度刻画。最后,与非线性评分函数的模型 CMN 和 NeuMF 相比,本文提出的新型混合评分函数结合现有推荐模型也能取得更好的推荐效果。

5.3 混合评分函数的相似度分布分析

正如第 4.1 节中所讨论的,混合评分函数的两个性质使其优于内积和欧氏距离。这里,本文通过分析 item 之间的相似性和距离分布来探究模型是如何从混合评分函数中获益的。为此,本文在 Sports 数据集上运行具有不同评分函数的 MF 模型。首先,为了验证 MixRec 在学习相似性方面的优势,对于 Sports 数据集中的每个用户,计算该用户交互过的所有物品之间的相似性。用最小-最大归一化方法对所有的相似性进行归一化,并计算相似性频率分布。图 2(a)~图 2(c)给出了使用不同评分函数后的相似性分布情况。从结果可以发现,与内积和距离评分函数相比,混合评分函数学到的物品与物品之间的相似性集中在值较大的区域。MixRec (MF)的大部分相似性都大于 0.5。这些结果验证了混合评分函数可以更有效地学习物品之间的相似性。

其次,由于在混合评分函数中用户和物品的最佳距离是非零的,因此 MixRec 不会把具有交互关系的用户和物品在潜在空间内优化成同一个点。为了验证这个性质,本文计算了已有交互中用户与物品之间的距离。图 2(d)~图 2(f)给出了欧氏距离的分布情况。对于欧氏距离评分函数,本文观察到用户与其交互过的物品的距离大部分小于 2.5 并且接近 0。而使用混合评分函数时,距离在 8 左右。这意味着混合评分函数并不会像欧氏距离函数那样简单地把用户和物品优化到同一个点上。此外,因为没有对距离进行优化,内积函数学习到的距离较大,距离集中在 12~14。

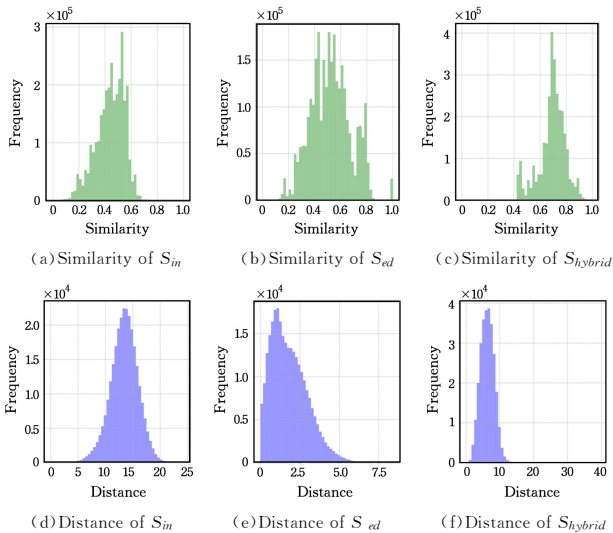


图2 相似度分布情况

Fig. 2 Distribution of similarities

注:图 2(a)~图 2(c)表示物品与物品之间相似度的分布,图 2(d)~图 2(f)表示用户与物品之间欧氏距离的分布。实验结果是在 Sports 数据集上使用不同评分函数的 MF 模型进行训练统计得到的

5.4 参数 γ 的影响

参数 γ 决定了混合评分函数中欧氏距离与内积的重要性,本文进行了实验以分析它对模型的影响。本文在 6 个数据集上运行 MixRec(MF),并使参数 γ 的值从 0 变化到 1,实验结果如图 3 所示。当 γ 等于 0 时,混合评分函数退化为内积,模型表现较差。随着 γ 的增加,即在评分函数中引入欧几里德距离,在 6 个数据集上模型性能都得到了显著提高。但当欧几里德距离的权重继续升高时,模型性能开始下降。特别是当 $\gamma=1$ 时(此时混合评分函数退化为欧氏距离评分函数),模型表现出了较差的准确率。这些结果表明,在内积和欧氏距离函数之间存在一种权衡,所提混合评分函数有助于通过混合它们来提供更好的推荐结果。

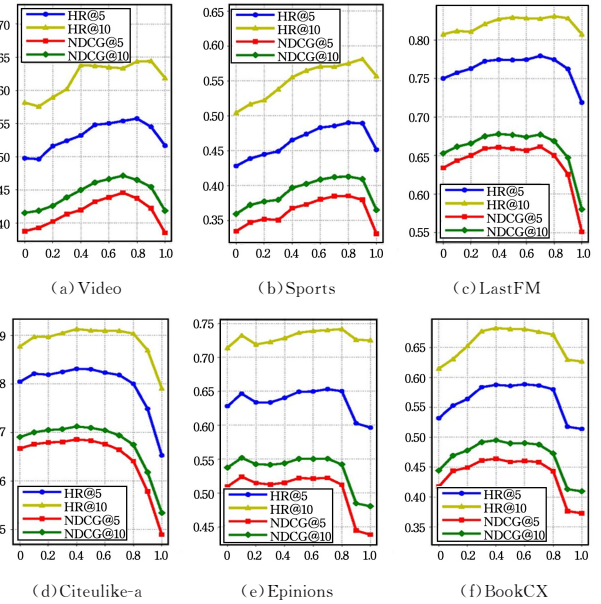


图3 不同 γ 下 MixRec(MF) 在各数据集上的实验结果

Fig. 3 Experimental results of MixRec(MF) with different γ on six datasets

结束语 评分函数是推荐模型的重要组成部分,对模型推荐的结果有着很大的影响。本文提出了一个新颖的混合评分函数,其可以有效地、灵活地学习用户与用户之间以及物品与物品之间的相似性,弥补了现有评分函数的不足。混合评分函数通过线性融合内积相似性和欧氏距离度量,有效整合了经典评分函数的优势,提高了推荐模型的准确性。本文不仅从理论上分析了新提出的评分函数的优点,而且在多个公开真实的数据集上进行了全面的实验验证。综上,混合评分函数是一种能够有效提升现有模型推荐效果,且简单有效的通用推荐评分技术。

参考文献

- [1] YI X, YANG J, HONG L, et al. Sampling-bias-corrected neural modeling for large corpus item recommendations[C]// Proceedings of ACM Conference on Recommender Systems, Copenhagen; Association for Computing Machinery, 2019: 269-277.

- [2] EBESU T, SHEN B, FANG Y. Collaborative Memory Network for Recommendation Systems[C]// Proceedings of the International ACM SIGIR Conference. Ann Arbor: Association for Computing Machinery, 2018: 515-524.
- [3] KOREN Y, BELL R, VOLINSKY C. Matrix Factorization Techniques for Recommender Systems[J]. Computer, 2009, 42(8): 30-37.
- [4] CHEN X, XU H, ZHANG Y, et al. Sequential Recommendation with User Memory Networks[C]// Proceedings of the ACM International Conference on Web Search and Data Mining. Marina Del Rey: Association for Computing Machinery, 2018: 108-116.
- [5] RENDLE S, FREUDENTHALER C, GANTNER Z, et al. BPR: Bayesian personalized ranking from implicit feedback[C]// Proceedings of the Conference on Uncertainty in Artificial Intelligence. Montreal: AUAI Press, 2009: 452-461.
- [6] ZHENG L, LU C, JIANG F, et al. Spectral collaborative filtering[C]// Proceedings of ACM Conference on Recommender Systems. Vancouver: Association for Computing Machinery, 2018: 311-319.
- [7] HE X, LIAO L, ZHANG H, et al. Neural Collaborative Filtering[C]// Proceedings of the International Conference on World Wide Web. Perth: International World Wide Web Conferences Steering Committee, 2017: 173-182.
- [8] HSIEH C, YANG L, CUI Y, et al. Collaborative Metric Learning[C]// Proceedings of the International Conference on World Wide Web. Perth: International World Wide Web Conferences Steering Committee, 2017: 193-201.
- [9] XIAO H, HUANG M, ZHU X. From one point to a manifold: knowledge graph embedding for precise link prediction[C]// Proceedings of the Conference on Uncertainty in Artificial Intelligence. New York: AUAI Press, 2016: 1315-1321.
- [10] WANG X, HE X, WANG M, et al. Neural Graph Collaborative Filtering[C]// Proceedings of the International ACM SIGIR Conference. Paris: Association for Computing Machinery, 2019: 165-174.
- [11] WANG H, WANG N, YEUNG D. Collaborative Deep Learning for Recommender Systems [C]// Proceedings of the ACM SIGKDD International Conference. Sydney: Association for Computing Machinery, 2015: 1235-1244.
- [12] KHOSHNESHIN M, STREET W. Collaborative filtering via euclidean embedding[C]// Proceedings of ACM Conference on Recommender Systems. Barcelona: Association for Computing Machinery, 2010: 87-94.
- [13] CEN Y, ZOU X, ZHANG J, et al. Representation Learning for Attributed Multiplex Heterogeneous Network[C]// Proceedings of the ACM SIGKDD International Conference. Anchorage: Association for Computing Machinery, 2019: 1358-1368.
- [14] MASSA P, AVESANI P. Trust-aware recommender systems [C]// Proceedings of ACM Conference on Recommender Systems. Minneapolis: Association for Computing Machinery, 2007: 17-24.
- [15] ZIEGLER C, MCNEE S, KONSTAN J, et al. Improving recommendation lists through topic diversification[C]// Proceedings of the International Conference on World Wide Web. Chiba: International World Wide Web Conferences Steering Committee, 2005: 22-32.
- [16] MCAULEY J, TARGETT C, SHI Q, et al. Image-Based Recommendations on Styles and Substitutes[C]// Proceedings of the International ACM SIGIR Conference. Santiago: Association for Computing Machinery, 2015: 43-52.
- [17] HE X, CHEN T, KAN M, et al. TriRank: Review-aware Explainable Recommendation by Modeling Aspects[C]// Proceedings of the ACM International Conference on Information and Knowledge Management. Melbourne: Association for Computing Machinery, 2015: 1661-1670.
- [18] HERLOCKER J, KONSTAN J, BORCHERS A, et al. An algorithmic framework for performing collaborative filtering[C]// Proceedings of the International ACM SIGIR Conference. Berkeley: Association for Computing Machinery, 1999: 230-237.
- [19] SARWAR B, KARYPIS G, KONSTAN J, et al. Item-based collaborative filtering recommendation algorithms [C]// Proceedings of the International Conference on World Wide Web. Hong Kong: International World Wide Web Conferences Steering Committee, 2001: 285-295.
- [20] KOREN Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model[C]// Proceedings of the ACM SIGKDD International Conference. Las Vegas: Association for Computing Machinery, 2008: 426-434.



XIAO Shi-tao, born in 1998, postgraduate, is a member of China Computer Federation. His main research interest is recommender system.



SHAO Ying-xia, born in 1988, Ph.D. research associate professor, is a member of China Computer Federation. His main research interests include large-scale graph analysis, parallel computing framework and knowledge graph analysis.