新并不总是更好：深度学习真的能从隐式反馈推荐中获益吗?

Yushun Dong
yd6eb@virginia.edu 弗
吉尼亚大学

Jundong Li
jundong@virginia.edu 弗
吉尼亚大学

Tobias Schnabel
toschnab@microsoft.com 微
软研究院

arXiv:2305.01801v1 [cs.IR] 2 May 2023

摘要

近年来，神经模型屡屡被誉为在推荐领域展现出最新水平的性能。然而，近期多项研究表明，许多神经推荐模型所报告的最新水平结果无法可靠复现。主要原因之一是，现有评估采用了各种不一致的协议。因此，这些可复现性问题使我们难以了解这些神经模型究竟能带来多少收益。由此可见，有必要对传统模型与神经模型进行公平、全面的性能比较。

受这些问题的启发，我们开展了一项大规模、系统性的研究，对基于隐式数据进行前n推荐的近期神经推荐模型与传统推荐模型进行比较。我们提出了一组评估策略，用于衡量推荐模型的记忆性能、泛化性能和子群体特定性能。我们在九个常用数据集上，对13种流行的推荐模型进行了广泛实验，其中包括两种神经模型和11种作为基线的传统模型。实验表明，即使进行了广泛的超参数搜索，神经模型也并未在所有方面都优于传统模型，例如其平均命中率表现更差。我们进一步发现，在某些方面，神经模型似乎优于非神经模型，例如推荐多样性，以及不同用户子群和物品子群之间的鲁棒性。我们的工作阐明了神经模型在推荐领域的相对优势与不足，因此是构建更优秀推荐系统的重要一步。

CCS 概念

• **通用与参考** → **评估**； • **信息系统** →**推荐系统**。

关键词

推荐系统、深度学习、评估

ACM 引用格式：

Yushun Dong、Jundong Li 和 Tobias Schnabel。2023。《新并不总是更好：深度学习真的能从隐式反馈推荐中获益吗？》。载于第*46*届 *ACM* 信息检索研究与发展国际会议论文集（*SIGIR '23*），*2023*年*7*月*23*日至*27*日，中国台湾台北。ACM，纽约，纽约州，美国，11页。https://doi.org/10.1145/3539618.3591785



本作品采用知识共享署名国际 4.0 许可协议授权。

SIGIR '23，*2023*年*7*月*23*日至*27*日，中国台湾台北
© 2023 版权所有归所有者/作者所有。ACM ISBN 978-1-4503-9408-6/23/07。
https://doi.org/10.1145/3539618.3591785

1 引言

近年来，基于深度学习的方法在推荐领域引发了广泛的研究兴趣，例如兴趣点（POI）推荐 [39, 81, 91], 、电子商务推荐 [15, 77, 86], 和新闻推荐 [23, 85, 96], 。基于深度学习的方法通过一系列神经网络层对用户偏好进行参数化，并在物品库 [41, 42, 60, 94]上优化排序性能。凭借强大的建模能力，神经模型理论上能够从隐式用户反馈中捕捉更复杂的非线性关系。进而，这应当提高找到高精度推荐模型的可能性。

尽管神经模型在许多其他领域取得了毋庸置疑的成功，但多项研究指出，许多神经推荐模型所报告的性能提升难以复现 [32, 33, 62]。例如，近期有研究发现，一些神经推荐模型的表现不如非常简单的推荐模型，例如线性模型 [33, 61]和基于最近邻的方法 [61]。造成此类可复现性问题的一个主要原因是，报告中的性能优势是在许多不同的实验设置下得出的 [19, 74, 94,95], ，例如数据集划分比例、评估协议、指标、推荐任务等。这引出了一个重要问题：神经模型能为推荐带来多大程度、何种类型的益处？这对从业者和系统设计者而言至关重要 [22]。要回答这一问题，我们需要更准确地了解神经推荐模型与非神经模型在何种情境下（即在什么场景中）以及以何种方式表现不同。

本文聚焦于探究神经推荐模型与非神经模型在 top-n 物品推荐这一最常见的推荐任务之一中的表现差异 [25]。尽管此前已有关于 top-n 推荐系统基准测试的研究 [2, 20, 76], ，但其中许多工作要么未纳入神经模型，要么仍有重要问题未得到解答。例如，近期一项同时分析神经模型和非神经模型的研究 [82, 83], 仍缺乏对这两类模型表现差异的全面分析，因此未能回答神经模型相较于传统模型可能有哪些优势。另一类研究只在有限的方面或设置下考察神经推荐模型与非神经推荐模型的表现差异 [33, 62] ，忽略了神经模型和非神经模型可能存在差异的其他方面。例如， [33] 只关注测试集上的排序准确率

。Ludwig 等人 [62] 比较了神经模型和非神经模型在基于会话的推荐任务中的表现，却忽略了受到广泛研究的 top-n 推荐场景。更重要的是，这些基准测试仅关注弱泛化 设置 [63, 64], ，也就是说，训练期间所有用户和物品都是已知的。这显然与现实应用形成鲜明对比：在实际应用中，推荐系统需要在部署后接纳新加入的用户 [50]。为了贴近更现实的场景，本文分析了强泛化设置下的表现，其中新用户属于测试集。

为了深入了解神经推荐模型能够带来哪些益处，本文对神经模型和非神经模型开展了大规模、系统性的研究，并探索了准确性之外的广泛模型属性。具体而言，我们提出对推荐模型的记忆能力和泛化能力进行实证评估。记忆能力是指模型记住训练期间见过的用户数据的能力，而泛化能力是指模型在测试阶段为新用户做出准确预测的能力。如上所述，我们采用强泛化 [63, 64]设置来进行泛化评估，其中用户被划分为互不重叠的训练集、验证集和测试集。请注意，记忆能力和泛化能力通常存在权衡，这在机器学习中也称为偏差-方差权衡 [3, 87]。稳健的推荐性能通常需要在二者之间取得谨慎的平衡 [16]。因此，我们结合这两个视角，并将第一个研究问题正式表述为（**RQ1**）神经模型和非神经模型在记忆能力和泛化能力方面有何差异？此外，大多数数据集都具有异质性，包含特征各异的物品和用户子群（例如，物品流行度和用户偏好）。在这种背景下，推荐模型在某个子群上的表现可能不错，在其他子群上的表现却未必理想 [24, 26, 28]。因此，评估神经模型和非神经模型差异的一个重要维度，是它们在特定用户或物品子群上的推荐性能。了解这一点有助于分析推荐模型的潜在局限，这是未来提出改进方案的重要一步。我们将第二个研究问题表述为（**RQ2**）神经模型和非神经模型在子群体特定性能方面有何差异？

为回答上述研究问题，我们提出了一系列实用且全面的评估策略。我们考察了 13 种流行的推荐模型（包括两种神经模型，以及 11 种作为基线的传统模型），并在九个常用的真实世界数据集上进行了大量实验。实验结果表明，即使超参数得到了充分优化，在大多数情况下，神经模型在命中率方面也无法胜过传统模型。不过，我们发现，神经模型相较于传统的非神经模型仍有一定优势。例如，神经模型在平均排名和推荐多样性方面表现更好，对物品间语义关系的映射更准确，并且在不同实例子群（即用户和物品）之间表现出更强的鲁棒性。

总而言之，我们的贡献主要有三方面：**(1) 实验设计**。 我们提出了一套全面的评估策略。据我们所知，我们的工作是迈出的第一步旨在评估神经模型与非神经模型在记忆和泛化方面的顶级n 推荐性能差异。**(2) 性能比较**。 我们选取了 13 种流行的推荐模型和九个常用数据集开展实证研究。我们进行了大规模研究，评估上述定义的各个维度上神经推荐模型与非神经模型之间的差异。**(3) 综合分析**。 我们根据实验结果，对神经和非神经推荐模型进行了全面分析。此外，我们还指出了未来研究的具体方向。

2 评估策略

通过实验，我们旨在回答本文中的以下两个研究问题：

研究问题 1： 在记忆和泛化任务的性能方面，神经模型与非神经模型有何不同？**研究问题 2：** 在子群体特定性能方面，神经模型与非神经模型有何不同？

下面我们介绍用于评估记忆、泛化和子群体特定性能的符号及策略。

2.1 符号

我们分别用花体字母（例如 $\mathcal{A}$）、粗体小写字母（例如 $\mathbf{a}$）和普通小写字母（例如 a）表示集合、向量和标量。在基于隐式用户反馈的推荐中，特定用户与物品之间的历史交互是二元的，即用户要么喜欢某个物品，要么两者之间的关系未知，因此可将其表示为一个集合。我们分别将推荐系统中的用户集合和物品集合表示为 $\mathcal{U} = \{U_1, \ldots\ldots, U_{|\mathcal{U}|}\}$ 和 $\mathcal{I} = \{I_1, \ldots\ldots, I_{|\mathcal{I}|}\}$。

在记忆设置中，我们将整个用户集合 $\mathcal{U}$ 视为训练期间可见的用户（即训练用户），记为 $\mathcal{U}_{trn}$。在强泛化设置中，我们将用户划分为三个互不重叠的集合，即训练用户集合 $\mathcal{U}_{trn}$（$\mathcal{U}_{trn} \subset \mathcal{U}$）、验证用户集合 $\mathcal{U}_{val}$（$\mathcal{U}_{val} \subset \mathcal{U}$）和测试用户集合 $\mathcal{U}_{tst}$（$\mathcal{U}_{tst} \subset \mathcal{U}$，且 $\mathcal{U}_{trn} \cup \mathcal{U}_{val} \cup \mathcal{U}_{tst} = \mathcal{U}$）。

2.2 记忆评估

记忆宽泛地描述了模型回忆训练期间见过的物品的能力 [16, 31, 38]。基于这一概念，我们将推荐中的记忆能力衡量为模型恢复训练期间见过的隐式用户反馈的表现。现在，我们将介绍两种不同的任务，用于探测特定推荐模型的记忆能力。这两种任务都衡量模型恢复训练期间见过的物品的能力，但实现这一目标的方式各不相同。

留一记忆任务。 在此任务中，我们调整了广泛采用的留一协议 [46, 47, 58, 65] 来评估记忆能力。与原始协议不同，我们不留出一个测试物品，而是选取一个训练期间见过的物品。具体而言，我们为用户 $\mathcal{U}_{trn}$ 中的每位用户随机选择一个隐式反馈条目。然后，在向 $\mathcal{U}_{trn}$ 中的用户推荐物品时，我们衡量模型对此条目的排名有多靠前。

重排序记忆任务。 该任务的设置与上述任务类似，但我们在这里关注的是恢复用户的全部 训练条目的性能。更具体地说，对于

对于 Utrn 中的每位用户，我们衡量在向该用户推荐物品时，其训练条目是否会排在前列。

总而言之，这两项任务都侧重于对训练用户集 Utrn 进行评估。第一种策略衡量推荐模型预测单个缺失（留出）物品的能力，而第二种策略则关注模型拟合后召回所有训练条目的效果。

2.3 泛化评估

泛化通常被定义为将学习到的底层数据模式应用于未见过的数据，以进行预测 [9, 14, 16, 53, 56, 92]。相应地，当我们提及推荐模型的泛化性能时，指的是模型在训练过程中未见过的用户-物品组合上进行准确预测的能力。值得注意的是，推荐模型的泛化能力具有重要的现实意义，因为这类模型通常需要为新用户进行预测 [50]。下面我们将讨论评估泛化能力的策略。

强泛化任务。 我们认为，与弱泛化相比，强泛化在实践中能更好地反映推荐模型的性能。具体而言，评估强泛化能力需要在新用户（即训练期间不可见的用户）和现有物品上测试模型 [63]。根据第 2.1 节介绍的划分方式，我们使用 Utrn 中的用户进行训练，并为每位用户留出一个物品。训练完成后，我们使用 Utst 中的用户（每位用户留出一个物品）评估推荐性能。

强泛化下的语义连贯性。 为了更深入地了解泛化性能，仔细考察模型对物品间语义的捕捉效果也至关重要 [90]。我们提出的策略基于这样一个直觉：对于仅与某个物品发生过一次交互的用户，如果模型能够识别并推荐与该交互物品在语义上相似的其他物品，就说明模型很好地捕捉了物品间语义。具体而言，我们提出生成一组虚拟用户 $\tilde{\mathrm{U}}$。这些虚拟用户中的每一位都只与 I 中的一个独特物品交互（因此 $|\tilde{\mathrm{U}}| = |\mathrm{I}|$）。对于每个虚拟用户 $\tilde{U}_i \in \tilde{\mathrm{U}}$，我们将生成的用户档案（即生成的一次用户-物品交互）视为已知的历史隐式反馈，并使用推荐模型为所有其他物品生成偏好分数。此外，我们收集所有物品的语义信息（例如人工标注的语义标签），作为辅助信息。对于$\tilde{}$与 U_i交互的物品，我们还计算其自身语义与所有其他物品语义之间的相似度分数（例如基于余弦相似度计算）。最后，我们计算$\tilde{U}_i$预测偏好与语义相似度之间的皮尔逊相关系数。我们提出将 U 上的平均皮尔逊相关系数值$\tilde{}$作为衡量推荐模型利用物品间语义效果的总体指标。

2.4 子群体特定性能评估

对于特定数据集，推荐模型在不同实例子群（例如用户和物品）上的表现可能有所不同。这些子群是根据某些实例级特征从原始数据集中划分出来的。例如，当用户根据某项标准被划分为活跃组和非活跃组时……

电子商务数据集中的大多数推荐模型可能会向活跃用户群体 [59, 72]提供质量显著更高的推荐结果。相应地，我们将子群体特定性能定义为“按照某些实例级特征对原始数据集进行划分后，各实例子群体上的性能表现”。事实上，热启动子群与冷启动子群之间的划分是研究最为广泛的划分方式之一 [1, 11, 43, 73, 89]。受此启发，我们从用户和物品两个角度提出了热启动子群与冷启动子群的划分方法。在本节中，我们将进一步讨论评估子群体特定性能的策略。

用户划分：活跃与非活跃。我们根据用户的交互总数，将用户划分为活跃用户子群体和非活跃用户子群体。我们旨在探究两个用户子群体之间的推荐质量差异（不同模型之间）。具体而言，我们从 Utst 中选取排名前 13 的活跃用户和排名前 $\frac{1}{7}$ 的非活跃用户，分别构建活跃用户集 Uact 和非活跃用户集 Uina。这里，我们提出通过确保两个用户集中的正向交互总数大致相同来确定 r 的值，从而保证公平比较。给定一个特定的推荐模型，我们分别评估其在 Uact 和 Uina 上的推荐性能。

用户划分：相似与不相似。为了以更细的粒度评估泛化能力，我们根据 Utst 中用户与 Utrn 中用户的隐式反馈相似度，将 Utst 划分为两组。具体而言，我们首先计算每个测试用户（即 Utst 中的用户）与每个训练用户（即 Utrn 中的用户）之间的相似度分数。随后，对于每个测试用户，我们截取其与这些训练用户之间排名前-l 的相似度分数，并计算这些截断后相似度分数的总和。其中，l 是控制相似度截断长度的参数。我们将排名前13 且前-l 相似度分数总和最大的测试用户视为相似用户。同理，我们将前13 个前-l 相似度分数总和最小的用户视为不相似用户。对于给定的推荐模型，我们分别评估其在这两组测试用户上的表现。

划分物品：头部与尾部。最后，我们探究神经推荐模型与非神经推荐模型在头部物品（与测试用户发生交互总数较多的热门物品）和尾部物品（交互次数较少的物品）上的推荐质量有何差异。具体而言，我们首先根据与这些物品发生交互的用户集合 Utst 中的用户总数，将物品集 I 划分为头部物品集 Ihead 和尾部物品集 Itail。为确保公平比较，这两个集合中的物品所占的交互总数通常应大致相同。为进行评估，我们根据测试用户的留出交互所关联的物品属于哪个集合，将测试用户划分为两个集合。对于给定的推荐模型，我们分别在这两组测试用户上评估其性能。

3 基准测试模型

如前所述，本文关注的泛化设置是强泛化（如第 2.3 节所述）。因此，在性能比较中，我们仅纳入天然支持强泛化设置的推荐模型。总体而言，我们将两个神经模型与

表 1：所采用数据集的统计信息。请注意，所呈现的统计信息是经过二值化和 5-core 筛选预处理后的相应结果。"Orig. Sparsity"表示预处理前的稀疏程度。

数据集	用户数	物品数	原始稀疏度	5-Core 稀疏性	#评分	#每个项目的评分	#每位用户的评分数
ml100k	938	1,008	6.30×10^{-2}	5.75×10^{-2}	54.4 K	54.0	58.0
lastfm	1,859	2,823	2.78×10^{-3}	1.36×10^{-2}	71.4 K	25.3	38.4
kuai	1,411	3,065	9.96×10^{-1}	5.01×10^{-2}	216.7 K	70.7	153.6
bookx	13,854	34,609	3.21×10^{-5}	1.09×10^{-3}	521.1 K	15.1	37.6
ml1m	6,034	3,125	4.47×10^{-2}	3.05×10^{-2}	57.44万	183.8	95.2
Jester	50,109	100	5.63×10^{-1}	2.03×10^{-1}	100万	10,172.0	20.3
Amazon-E	124,895	44,843	3.91×10^{-6}	1.92×10^{-4}	110万	23.9	8.6
ml20m	136,674	13,680	5.40×10^{-3}	5.34×10^{-3}	10.0 M	729.3	73.0
netflix	463,435	17,721	1.18×10^{-2}	6.93×10^{-3}	56.9 M	3,209.7	122.7

五类非神经推荐基线进行比较：非个性化模型、基于因子分解的模型、基于最近邻的模型、线性模型和基于图的模型。

神经推荐模型。 只有少数神经推荐模型能够在强泛化设置下生成推荐结果。我们选择了 MultiVAE 和 MultiDAE，因为它们都很常见，而且在某些数据集上达到了（接近）最新水平的性能 [60]。MultiVAE 和 MultiDAE 都是基于自编码器的推荐模型，其基本思路是根据用户画像的不完整版本重建完整的用户画像。在 MultiDAE 的基础上，MultiVAE 采用完全贝叶斯方法来拟合每位用户的方差，这可能会使其更容易过拟合。

非个性化模型（非神经网络）。 采用了两种非个性化推荐模型作为基线，即随机和流行度。随机模型为每位用户生成物品的随机排列，而流行度模型则根据物品的流行度（正向交互的总数）为用户生成推荐结果。

基于因子分解的模型（非神经网络）。 选取了两种基于因子分解的模型，包括 PureSVD 和交替最小二乘。二者的主要区别在于，PureSVD 对用户-物品矩阵执行普通奇异值分解，而交替最小二乘执行加权矩阵分解。

基于最近邻的模型（非神经网络）。 采用 ItemKNN 和 UserKNN 作为两种传统的基于最近邻的推荐模型。在这两种模型中，推荐结果分别基于物品间相似度和用户间相似度得出。

线性模型（非神经网络）。 SLIM [68] 和 Ease [80] 是两种常用的线性基线。二者都旨在学习一个线性函数，以捕捉基于物品的协同过滤中的相似性。然而，SLIM 旨在学习稀疏线性函数，而 Ease 不施加稀疏性约束。因此，Ease 采用可学习权重矩阵的弗罗贝尼乌斯范数作为正则化项，而不是像 SLIM 那样使用 l_1-范数。

基于图的模型（非神经网络）。 P3alpha [17], RP3beta [69], 和基于图滤波器的协同过滤（GFCF） [79] 被用作基于图的推荐模型。通常，这些模型将用户和物品视为节点，相应的输入矩阵描述了用户与物品之间已有的边以及隐式用户反馈。具体而言，在 P3alpha 中，根据三步游走到达每个用户的概率对物品进行排序，以供推荐；RP3beta 是 P3alpha 的改进版本，其结果会按照相应物品的流行度进行归一化；GFCF 则基于沿现有边传播后的隐式用户反馈输入矩阵进行预测。

4 实验设置

4.1 数据集

数据集选择。 许多先前的研究只在相对少量的数据集上发布结果 [83]。我们选择数据集的指导原则是选用最常用的公开数据集，以确保可复现性。本文选取了九个广受欢迎的公开数据集，分别是 ml100k [40],ml1m [40], ml20m [40], lastfm [45], kuai [35], bookx [12],Jester [36], Amazon-E [66], 和 Netflix [10]。

预处理策略。 我们基于隐式用户反馈（例如表示点击和购买的二元条目）开展实验，并将所有有序评分数据转换为二元数据。我们遵循普遍共识，将所有大于或等于四分的评分视为正向评分，其余评分视为负向评分 [83]。此外，在实际应用中，通常还需要过滤掉评分不足的用户和物品，即执行 h-核心过滤。我们选择一种常见设置 [94]，将 h 设为五，并在表 1 中列出过滤后的统计信息。

4.2 指标

本小节介绍我们选择的评估指标。具体而言，我们纳入了三类指标，分别用于衡量效用、多样性和语义连贯性。详情如下。

效用指标。 在 top-n 推荐中，已有多种效用指标用于评估排序性能。然而，在评估过程中，多种常用指标可能并不稳定。例如，nDCG 会受到物品总数的影响，仅改变计算中使用的增益因子，就可能颠覆模型比较的结论 [13]。在此，我们采用三种更稳定、常用的指标来衡量推荐效用，即 Recall@k、HitRate@k,

以及平均排名 [34]。我们将用户 U_i 的 Recall@k 定义为

$$\text{Recall@k} = \frac{|R(U_i) \cap T(U_i)|}{|T(U_i)|}, \quad U_i \in \mathcal{U}, \tag{1}$$

其中，$R(U_i)$ 表示用户 U_i 的前 n 个推荐结果中检索项的集合；$T(U_i)$ 表示测试阶段用于用户 U_i 的相关项集合。为与先前的研究保持一致，当 $|T(U_i)| = 1$ 时，我们使用 Recall@k 而非 HitRate@k。其公式如下：

$$\text{HitRate@k} = |R(U_i) \cap T(U_i)|, \quad U_i \in \mathcal{U}^*. \tag{2}$$

这里，U^* 分别表示记忆和泛化设置中的 U 和 Utst。最后，我们将平均排名定义为

$$\text{MeanRanks} = \frac{\sum_{I_i \in T(U_i)} \text{Rank}(I_i)}{|T(U_i)|}, \quad U_i \in \mathcal{U}^*, \tag{3}$$

其中，函数 Rank(·) 接收一个项目作为输入，并输出该项目在推荐结果中的对应排名位置。我们使用 Recall@k 和平均排名来衡量模型在重排序记忆任务下的效用。在其他设置中，我们使用 HitRate@k 和平均排名作为相应的指标。

多样性指标。多样性是最受欢迎的超越准确性排名指标之一 [6, 29, 44, 71, 84, 93]。在我们的实验中，我们采用基尼指数 [88] 和香农熵 [37] 来衡量多样性。基尼指数源自经济学，最初用于衡量不平等程度。计算时，统计某物品出现在测试集中用户的前-n 个推荐结果中的次数，然后对该频率进行归一化，使物品频率构成一个概率分布（例如，物品 I_i对应的 $p(I_i)$）。按递增顺序排列物品后 i_1，i_2，……，i_n，基尼指数定义为

$$\text{Gini_Index} = \frac{1}{n-1} \sum_{j=1}^{n} (2j - n - 1) p\left(I_{i_j}\right). \tag{4}$$

作为第二种多样性度量，我们采用香农熵。其计算方式如下：

$$\text{Shannon_Entropy} = -\sum_{i=1}^{n} p(I_i) \log p(I_i). \tag{5}$$

我们使用上述两种指标来衡量多样性，以便更准确地了解不同模型之间的性能差异。

语义一致性指标。神经模型与非神经模型的另一个重要差异在于，模型在多大程度上捕捉了嵌入空间中物品之间的语义关系（另见第 2.3 节）。受以用户为中心的研究（如 [90], ）启发，我们提出了一种名为语义一致性指数（SCI）的指标。对于每个物品 $I_i \in \mathrm{I}$，我们根据体现人类对相似性和关联性认知的信息（例如人工标注的语义标签）构建一个向量表示。令 r_i 为模型生成的项目相似度向量。对于使用项目-项目矩阵的模型，我们直接使用对应的矩阵元素。对于不具备此类矩阵的模型，我们输入一个独热用户向量，并将其第 i 个元素设为 1。我们还计算 I_i 的语义向量与每个其他物品的语义向量之间的余弦相似度，并将所得向量记为 s_i。随后，SCI 计算如下：

$$\text{SCI} = \frac{1}{|\mathcal{I}|} \sum_{I_i \in \mathcal{I}} \text{Pearson}\left(\mathrm{s}_i, \mathrm{r}_i\right), \tag{6}$$

其中，函数 f 是推荐模型，它以两个物品为输入，并输出一个向量，表示对 I 中所有物品的预测偏好分数；函数 Pearson(·) 以两个向量为输入，并输出它们的皮尔逊相关系数。我们利用 SCI 来衡量推荐模型对物品间语义的捕捉能力。

4.3 实验设置

评估协议。对于每个模型，我们采用包含五个基于用户的折的嵌套交叉验证进行训练和测试。每轮评估中，我们使用其中三折作为训练数据，一折用于验证，一折用于测试。我们将测试集与训练集用户相似度排序截断（子群体特定性能评估下）的截断长度设为十 l。

超参数调优。我们对所有模型进行了广泛的超参数搜索。具体而言，在每一轮交叉验证中，我们使用贝叶斯优化进行了 50 次迭代的超参数搜索，并将 HitRate@50 作为目标指标。每轮交叉验证中各数据集的搜索空间和最优超参数值将在论文录用后公开。

5 实证研究

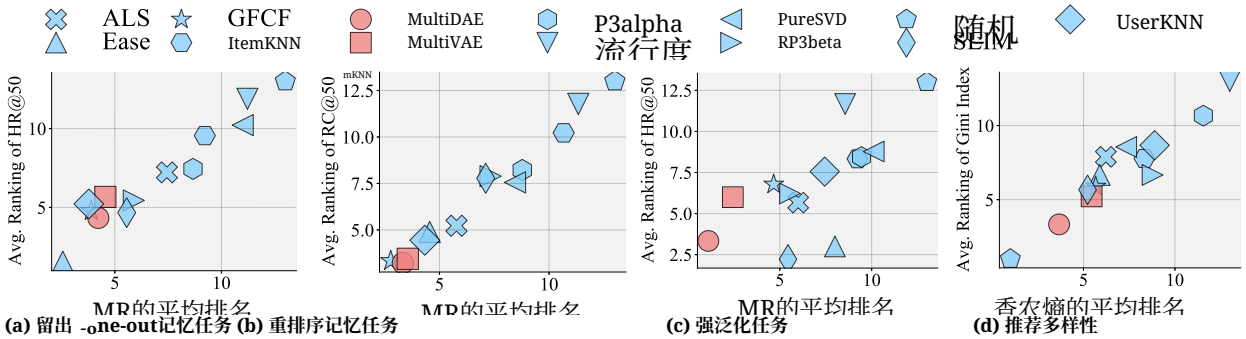
5.1 发现 1：神经模型在规模较大的数据集上更擅长记忆

我们首先在第 2.2 节介绍的两种记忆设置上进行实验。在留一记忆任务中，我们收集 HitRate@50 和平均排名的数值。在重排序记忆任务中，我们收集召回率@50 和平均排名的数值。我们计算每个数据集上所有模型的排名，然后根据它们在所有数据集上的平均排名进行排序。结果见图 1(a) 和图 1(b)。

留一记忆任务。我们首先讨论神经模型与非神经模型在留一记忆任务中的差异。从实验结果可以观察到，神经模型并未表现出优于其他非神经模型的优势，而且更简单的非神经模型仍然能够超越神经模型。例如，作为一种线性推荐模型，Ease 在两项效用指标上均优于神经模型。

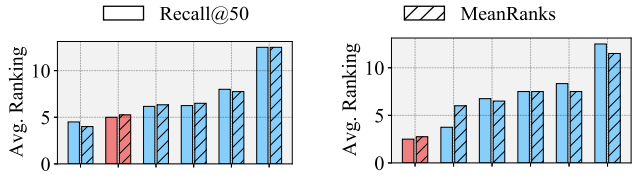
重排序记忆任务。我们考察了神经模型与非神经模型在所谓的重排序记忆任务中的差异。我们观察到，神经模型属于重排序性能最佳的模型之列。一个可能的原因是，较高的模型容量使神经模型能够很好地记忆训练过程中见过的所有隐式用户反馈。尽管如此，值得注意的是，若干非神经模型也表现出色，例如线性模型，这与弱泛化设置下现有研究（例如 [33]）的发现一致。

不同数据集规模下的记忆。基于前面的结果，我们进一步深入探究神经模型与非神经模型在拟合能力方面的差异。具体而言，我们针对每个基线模型，计算其在重排序记忆任务中，在规模最小和最大的 top-k 个数据集上的平均表现排名（以隐式用户反馈总数衡量）。我们展示了 $k = 2$ 的结果



(a) 留出 -one-out记忆任务 (b) 重排序记忆任务 (c) 强泛化任务 (d) 推荐多样性

图 1：在 (a) 和 (b) 中，我们分别展示了两种不同记忆任务的结果。在 (c) 中，我们从推荐效用的角度展示了泛化结果。在 (d) 中，我们以基尼指数和香农熵为指标，展示了推荐多样性的结果。我们分别使用"RC"、"HR"和"MRs"表示召回率、命中率和平均排名。对于每个推荐模型，平均排名均在全部九个数据集上计算得出。两种神经推荐模型以浅珊瑚色表示，其他模型则以浅蓝色表示。



(a) 在规模最小的两个数据集上的平均排名（两个规模最小的数据集） (b) 在规模最大的两个数据集上的平均排名（两个规模最大的数据集）

图 2：神经模型与其他五组基线在重排序记忆任务中的结果。每组基线的性能取两个最小/最大数据集上的平均值。两个神经模型以浅珊瑚色表示，其他模型以浅蓝色表示。

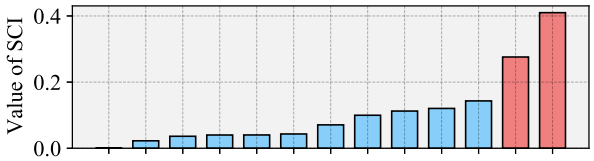


图3：ml20m 数据集上的语义连贯性结果。两个神经推荐模型以浅珊瑚色标示，其余模型以浅蓝色标示。

如图 2(a) 和图 2(b) 所示。我们观察到，在规模最小的前-k 个数据集上，线性推荐模型的性能优于所有其他模型。然而，在规模最大的前-k 个数据集上，神经推荐模型取得了最佳性能。尽管未在图中展示，对于 k的其他取值，这些趋势同样成立。一种解释是，隐式用户反馈较少（即用户-物品交互较少）的小型数据集往往具有更简单的用户偏好模式。因此，线性模型等更简单的模型能够较好地记忆用户档案。另一种解释可能是，由于神经模型具有更高的容量，因此需要更多数据才能避免过拟合。

总结这些观察结果后，我们发现神经模型在留一记忆方面并未表现出更优的性能。但它们在重排序记忆任务中却属于表现最佳的模型之一。我们假设，神经模型在重排序任务中表现更好，是因为该任务强调同时正确恢复所有训练条目的性能。凭借其平滑的参数化方式，神经模型可能比稀疏线性模型更擅长实现这种软聚类。

5.2 研究发现 2：神经模型与非神经模型的泛化方式不同

接下来，我们将探讨神经模型与非神经模型在泛化能力方面的差异。除了单纯的准确率之外，我们还将从三个角度讨论它们之间的差异，包括推荐效用、多样性和语义连贯性。

推荐效用。 这通过在强泛化设置下测量命中率（HitRate）和平均排名（MeanRanks）来评估推荐效用。我们首先在图 1(c) 中展示各推荐模型排名的总体概况。我们观察到，就 HitRate@50（纵轴）而言，尽管神经模型属于性能最佳的模型之列，但两种线性模型（Ease 和 SLIM）的表现均优于神经模型。换句话说，神经模型的表现并不比传统线性模型更好。然而，当通过测量平均排名（MeanRanks）（横轴）来考察相关项在排名中的位置时，神经模型优于所有其他推荐模型。这些实验结果再次表明，神经模型能够形成软且连贯的簇，这与重排序记忆任务中的表现类似。

此外，我们还在表 2 中呈现了详细的性能统计数据。这些详细结果与我们之前的总结一致：神经模型在平均排名方面往往优于其他非神经模型，但在命中率方面并不比线性模型表现出任何优势。

推荐多样性。 为了研究推荐多样性方面的性能差异，我们计算了所得推荐结果的基尼指数和香农熵。为获得更强的区分能力，我们考察前十（$k = 10$）的性能分数，并在图 1(d) 中展示了各模型的多样性排名。我们的观察如下。首先，不出所料，

非个性化的随机方法在推荐多样性的两项指标上都取得了最佳表现。然而，由于效用较差（例如图 1(c) 所示的命中率和平均排名较低），随机方法并不是适用于现实应用的有效推荐方法。在其余模型中，神经模型在两项多样性指标上的表现最佳。这些结果表明，神经模型给出的推荐结果具有更高的多样性，这意味着神经模型不太容易给热门物品较高的分数。

推荐的语义连贯性。最后，我们分析了推荐的语义连贯性，并使用从用户指定标签中收集的物品辅助信息。此处展示的结果基于 MovieLens ml20m 数据集，其中的人工标注语义信息来自用户提交的语义标签。我们遵循第 2.3 节所述的方法，并在图 3 中展示实验结果。我们发现，神经模型的语义一致性指数最高。这一观察结果表明，神经模型在从用户反馈中学习时，更善于捕捉物品间语义。

总而言之，我们发现，在强泛化设置下，非神经模型（平均而言）的命中率更高，而神经模型的平均排名表现更好。这至少为从业者提供了一些指导：如果希望获得更高的命中率，应优先选择非神经模型。此外，与非神经模型相比，神经模型的多样性更高，也更善于捕捉物品间的语义关系。因此，当多样性和充分的物品间关系十分重要时，应优先选择神经模型。

5.3 发现三：神经模型在不同子群中表现出更强的鲁棒性

现在，我们将探讨第 4 节中所述的神经模型与非神经模型之间的子群体特定性能差异。与之前一样，我们考察强泛化设置下不同用户子群体和物品子群体之间的性能差异。

相似用户与不相似用户。我们首先研究相似用户与不相似用户之间的性能差异。这里
相似用户是指这样的测试用户：与训练期间已观察到的用户（就隐式用户反馈而言）相比，其余弦相似度排名最高；而不相似用户则是与这些训练用户最不相似的用户。我们分别在图 4(a) 和图 4(b) 中展示了这两个用户子群体的实验结果。我们得出以下观察结果。首先，神经模型和非神经模型呈现出与泛化实验中所发现的排名相似的性能（见第 5.2 节）：神经模型在平均排名方面表现更好，但在命中率方面无法超越传统的线性推荐模型。其次，与其在相似用户上的性能相比，MultiDAE 在不相似用户上的表现更好。其他模型在命中率方面没有明显变化。然而，就平均排名而言，在不相似用户上，两种神经模型的表现都显著优于非神经模型。这些发现表明，在两个不同用户子群体（即相似用户和不相似用户）之间，无论是命中率还是平均排名，神经模型的性能都更加稳健。

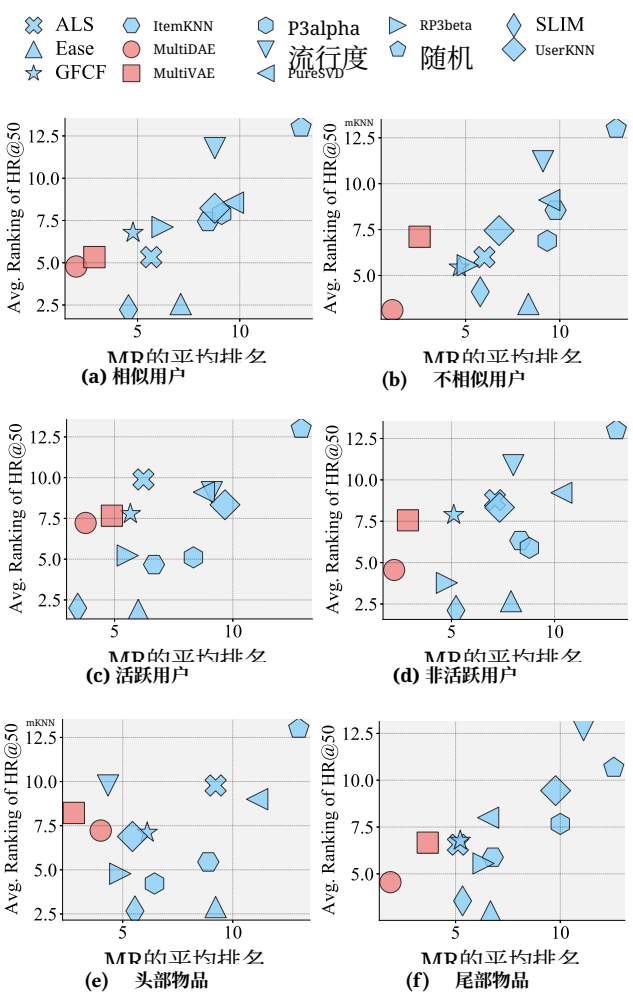


图 4：子群体特定性能评估。我们分别用“HR”和“MRs”表示命中率和平均排名。两个神经模型以浅珊瑚色标出，其他模型则以浅蓝色标出。

活跃用户与非活跃用户。我们还分析了活跃用户和非活跃用户之间的性能差异。这里，活跃用户指评分数量最多的用户，而非活跃用户则相反（阈值根据百分位数选取）。我们的发现如下。首先，对于活跃用户子群体，两种神经模型在平均排名方面的表现都优于命中率方面的表现。不过，它们并未始终优于非神经模型。例如，SLIM 是一个简单得多的非神经模型，但在两项效用指标上的表现都优于神经模型。其次，对于非活跃用户，MultiDAE 的表现更好，而且大多数模型的命中率表现保持稳定。然而，大多数非神经模型在非活跃用户上的平均排名表现有所下降，而两种神经模型的表现最佳。这些观察结果表明，与非神经模型相比，神经模型在不同活跃程度的用户之间展现出更好的鲁棒性（尤其是在平均排名方面）。

(a) ml100k。

模型	HR@50	95%置信区间	平均排名	95%置信区间
随机	0.047	0.030	476.3	39.5
流行度	0.324	0.067	193.5	30.1
UserKNN	0.553	0.071	107.3	23.8
ItemKNN	0.569	0.071	102.9	22.0
P3alpha	0.570	0.071	111.3	24.8
RP3beta	0.571	0.071	95.4	19.8
PureSVD	0.577	0.071	108.7	24.9
MultiVAE	0.579	0.071	92.1	19.6
GFCF	0.582	0.071	97.7	21.6
MultiDAE	0.597	0.070	85.8	18.6
ALS	0.598	0.070	96.7	22.0
Ease	0.600	0.070	99.1	23.9
SLIM	0.604	0.070	101.5	23.2

(b) ml1m。

模型	HR@50	95%置信区间	平均排名	95%置信区间
随机	0.018	0.007	1518.1	50.0
流行度	0.251	0.024	410.6	29.1
ItemKNN	0.441	0.028	276.3	30.2
UserKNN	0.449	0.028	258.4	29.0
P3alpha	0.456	0.028	296.3	31.6
RP3beta	0.462	0.028	242.1	26.3
PureSVD	0.471	0.028	271.1	32.3
GFCF	0.487	0.028	219.9	25.6
MultiVAE	0.492	0.028	180.3	18.5
SLIM	0.496	0.028	208.4	23.1
MultiDAE	0.503	0.028	173.8	18.1
ALS	0.504	0.028	216.8	25.6
Ease	0.512	0.028	224.4	27.1

(c) ml20m。

模型	HR@50	95%置信区间	平均排名	95%置信区间
随机	0.004	0.001	6814.6	46.6
流行度	0.241	0.005	695.9	14.7
P3alpha	0.432	0.006	957.8	24.9
ItemKNN	0.462	0.006	559.9	18.4
PureSVD	0.469	0.006	808.5	30.9
RP3beta	0.476	0.006	374.7	12.6
UserKNN	0.496	0.006	517.4	18.6
GFCF	0.516	0.006	385.9	17.5
ALS	0.519	0.006	494.7	22.2
MultiVAE	0.522	0.006	253.8	7.8
SLIM	0.523	0.006	405.1	14.8
Ease	0.548	0.006	744.3	30.4
MultiDAE	0.564	0.006	197.7	6.6

（d）lastfm。

模型	HR@50	95%置信区间	平均排名	95%置信区间
随机	0.024	0.015	1374.2	82.2
流行度	0.206	0.041	660.6	78.9
PureSVD	0.464	0.051	337.8	64.7
UserKNN	0.467	0.051	311.0	60.4
P3alpha	0.491	0.051	220.0	44.0
ALS	0.492	0.051	295.4	58.9
ItemKNN	0.505	0.051	193.2	37.8
RP3beta	0.508	0.051	191.5	38.1
Ease	0.522	0.051	287.5	59.0
GFCF	0.527	0.051	205.0	43.3
SLIM	0.530	0.051	201.6	41.4
MultiVAE	0.540	0.051	172.3	34.6
MultiDAE	0.544	0.051	167.7	33.9

(e) kuai。

模型	HR@50	95%置信区间	平均排名	95%置信区间
随机	0.016	0.015	1419.9	98.9
P3alpha	0.318	0.054	601.7	90.2
MultiVAE	0.319	0.054	567.4	84.7
流行度	0.319	0.054	588.9	86.0
GFCF	0.320	0.054	590.4	87.6
ItemKNN	0.322	0.055	621.6	90.8
RP3beta	0.322	0.055	578.9	87.2
MultiDAE	0.323	0.055	577.3	84.9
UserKNN	0.323	0.055	598.6	87.7
PureSVD	0.327	0.055	628.8	93.6
ALS	0.330	0.055	599.8	90.7
Ease	0.330	0.055	617.3	92.7
SLIM	0.331	0.055	607.4	89.0

(f) bookx.

模型	HR@50	95%置信区间	平均排名	95%置信区间
随机	0.001	0.001	17270.8	372.2
流行度	0.055	0.008	9118.9	395.9
PureSVD	0.109	0.012	10701.4	450.7
ALS	0.117	0.012	9704.5	430.6
MultiVAE	0.129	0.012	4976.9	275.0
GFCF	0.132	0.013	4852.8	282.9
MultiDAE	0.137	0.013	4704.4	262.9
UserKNN	0.137	0.013	7749.4	373.8
ItemKNN	0.145	0.013	10098.6	400.6
P3alpha	0.153	0.013	8949.8	392.8
Ease	0.156	0.013	10142.6	463.3
RP3beta	0.157	0.014	7526.9	374.7
SLIM	0.161	0.014	8243.6	376.2

(g) Jester。

模型	HR@50	95%置信区间	平均排名	95%置信区间
随机	0.633	0.009	40.8	0.5
PureSVD	0.837	0.007	23.6	0.5
流行度	0.891	0.006	20.0	0.4
ItemKNN	0.912	0.006	18.3	0.4
P3alpha	0.927	0.005	17.4	0.4
GFCF	0.928	0.005	17.1	0.4
ALS	0.936	0.005	15.4	0.3
Ease	0.939	0.005	14.9	0.3
MultiVAE	0.939	0.005	15.7	0.3
RP3beta	0.939	0.005	16.0	0.3
SLIM	0.939	0.005	15.1	0.3
UserKNN	0.940	0.005	15.4	0.3
MultiDAE	0.947	0.004	14.7	0.3

（h）amazon-e.

模型	HR@50	95%置信区间	平均排名	95%置信区间
随机	0.001	0.000	22423.5	160.7
流行度	0.063	0.003	9779.2	153.7
PureSVD	0.070	0.003	12358.1	179.7
ALS	0.089	0.004	12863.8	187.2
GFCF	0.114	0.004	10004.1	165.5
MultiVAE	0.114	0.004	7169.4	128.3
UserKNN	0.119	0.004	14265.9	178.9
MultiDAE	0.126	0.004	6424.7	120.2
Ease	0.128	0.004	17552.1	221.5
ItemKNN	0.128	0.004	15762.3	178.8
P3alpha	0.129	0.004	14208.5	178.4
RP3beta	0.129	0.004	14341.7	178.4
SLIM	0.132	0.004	12844.7	174.0

(i) Netflix。

模型	HR@50	95%置信区间	平均排名	95%置信区间
随机	0.003	0.000	8794.4	32.7
流行度	0.172	0.002	1005.6	12.2
P3alpha	0.361	0.003	1759.7	25.7
ItemKNN	0.374	0.003	1426.6	23.0
UserKNN	0.387	0.003	1144.4	20.4
RP3beta	0.389	0.003	1415.7	23.0
PureSVD	0.398	0.003	1281.5	24.2
GFCF	0.404	0.003	556.6	12.3
MultiVAE	0.417	0.003	420.9	6.8
ALS	0.424	0.003	742.4	16.5
SLIM	0.450	0.003	830.3	17.2
MultiDAE	0.462	0.003	327.8	5.8
Ease	0.467	0.003	1099.6	23.0

表 2：所有推荐模型在九个真实世界数据集上的性能。每个数据集上的结果按照 HitRate@50（记为 HR@50）从上到下升序排列。"95%置信区间"表示 95% 置信区间。神经模型以灰色标记。

头部物品与尾部物品。 最后，我们比较推荐模型在头部物品和尾部物品上的表现。同样，头部物品指隐式用户反馈数量最多的物品，而尾部物品则相反。我们根据随机留出的物品类型，将用户划分为两个子群体，并报告各模型在这两个子群体上的表现。我们发现，在命中率方面，神经模型并不优于非神经模型，尤其是对于留出物品为头部物品的用户。不过，在两种设置下，神经模型在平均排名方面确实具有优势。其次，在尾部物品上，两种神经模型都保持着相较于其他非神经模型的性能优势，而大多数其他非神经模型在尾部物品上的表现则有所下降。一些非神经模型（如交替最小二乘和 PureSVD）在尾部物品上的表现更好，但仍不及神经模型。这些观察结果表明，神经模型不仅通常可获得更高的平均排名分数，同时在头部和尾部物品上也表现出更好的鲁棒性。

基于以上讨论，我们的结果表明，与非神经模型相比，神经模型在热启动/冷启动实例上往往具有更强的鲁棒性。因此，如果实践中需要更好的平均排名或鲁棒性，我们建议优先选择神经模型。

6 相关工作

研究人员早已注意到，由于基线实现、数据集预处理策略 [76]，以及评估方法 [52]，存在差异，多种推荐模型的结果难以进行比较 [27]。这些发现使人们开始怀疑，每个新模型所报告的性能提升

是否真的能够累积起来。这促使研究人员重新审视现有的推荐模型 [4, 33]。大多数复现实验研究发现，很难获得与先前已发表结果相似的结果 [19]。这进一步凸显了这样一个观点：推荐系统中的可复现性和可复现性是核心问题[4, 8, 33, 62, 74]。例如，Sedhain 等人 [78] 指出，简单的线性分类模型与最新水平相比表现相当出色；Pérez 等人 [70] 发现，若干神经推荐模型无法实现可复现；Kolesnikov 等人 [51] 指出，广泛采用的评估策略存在潜在缺陷。

神经推荐模型与非神经推荐模型。近年来，有观点认为，与非神经模型相比，神经模型的表现优于传统推荐系统。据我们所知，目前只有另外三项研究探讨了神经模型与非神经模型之间的差异。Anelli 等人 [4]发现，神经模型无法胜过大多数传统模型，例如线性模型和矩阵分解。Ludewig 等人 [62] 在基于会话的推荐中也得出了类似结果。最后，Zhao 等人 [94] 指出，神经模型的训练效率显著低于其他传统模型，例如线性模型。这进一步令人怀疑，在实际应用中采用神经模型是否值得。

推荐系统评估。 值得注意的是，推荐系统并非孤立存在，而是面向用户的系统的一部分 [18]。在真实世界的在线推荐系统中，任何潜在缺陷都可能对用户造成巨大负面影响。因此，在用户接触推荐模型之前，系统地了解其特性至关重要。多项研究尝试对现有推荐模型进行系统评估 [4, 8, 33, 62, 74]。例如，Sun 等人 [83] 指出，某些关键因素（如数据划分方法和超参数调优策略）会显著影响推荐模型的性能。因此，他们提出了标准化流程，以便进行更严格的评估。近期其他研究也考察了整个评估流程中的其他环节 [5, 6, 33, 49, 55, 62]。此外，现有研究也关注评估中的特定因素。**(1)数据集构建**。 数据集对于现有推荐模型提供准确的推荐结果一直发挥着关键作用。数据集的预处理和构建策略受到广泛关注 [55, 57, 67, 75, 94]。例如，Sachdeva 等人 [75] 评估了十六种不同采样策略在推荐模型基准测试中的效果。**(2)基线**。 评估中是否纳入恰当的基线，也是推荐系统评估的一个关键因素。例如，Ji 等人 [48] 提议重新审视推荐系统中常用的基线。**(3) 模型优化**。 模型优化也会显著影响推荐系统的性能，例如超参数调优策略。比如，Zhao 等人 [94] 指出，超参数的搜索范围往往会显著影响性能；而 Anelli 等人 [7] 则指出，通常只需调优少数几个参数，就能取得足够好的性能。**(4) 评估指标**。 研究所采用的评估指标是否恰当也很重要 [21, 30, 54, 55, 94]。例如，Zhao 等人 [94] 发现，采用基于采样的指标可能会在评估过程中引入偏差。

我们工作的意义。尽管评估推荐模型方面已有进展，但比较不同类型推荐模型的研究并不多 [49, 62]。然而，我们认为这是一个关键问题，因为了解不同模型之间的差异有助于研究人员和工程师针对不同应用场景选择恰当的模型。与此同时，关于神经模型与非神经模型差异的现有见解也十分有限。例如，Anelli 等人 [4] 评估了神经模型与非神经模型。然而，他们采用了弱泛化作为评估策略；我们认为，这无法代表真实世界中需要向新用户提供推荐结果的推荐系统。Ludewig 等人 [62] 比较了神经模型与非神经模型的性能。然而，他们的研究仅关注基于会话的推荐，忽略了最常见的隐式用户反馈推荐问题。Zhao 等人 [94] 在考虑不同因素的情况下评估了热门推荐模型。然而，他们没有全面讨论这两类推荐模型的性能差异。与大多数现有的推荐模型评估研究不同，本文提供了强泛化设置下的全面评估结果，从而更深入地理解这两类推荐模型的特性。

7 结论与讨论

在本研究中，我们全面考察了神经推荐模型与非神经推荐模型之间的性能差异。我们通过实证研究，探讨了在实际推荐中使用神经模型而非传统的非神经模型可能带来哪些收益。我们提出了多种实用且多样的评估策略，用于基准测试；随后，在 9 个公开可用的真实世界数据集上，对 13 种热门推荐模型进行了广泛实验。结果表明，在大多数情况下，若以命中率（HitRate）为评估指标，神经模型的表现并不优于其他非神经模型；但在平均排名（MeanRanks）方面，神经模型表现更佳。此外，神经模型的平均多样性更高，更善于捕捉物品间语义，并且在热启动和冷启动场景下表现出更强的鲁棒性。

据我们所知，这是首篇研究在基于隐式用户反馈的 top-n 推荐任务中，神经推荐模型与非神经推荐模型之间性能差异的工作。研究涵盖了受到广泛关注的记忆能力、尚未得到充分探索的泛化能力，以及鲜少讨论的特定子群体性能。未来的研究可以基于本文进一步探讨神经模型是否在其他常用指标上也表现出优势，以及神经模型带来的收益是否值得其训练成本。然而，我们指出，这些问题超出了本文的研究范围。我们希望这项工作能够帮助从业者选择推荐模型，激发更多关于理解推荐模型的研究，并推动未来设计出更优秀的模型。

8 致谢

Yushun Dong 和 Jundong Li 获得美国国家科学基金会通过项目编号 IIS-2006844、IIS-2144209 和 IIS-2223769 提供的资助。

参考文献

[1] Nor Aniza Abdullah、Rasheed Abubakar Rasheed、Mohd Hairul Nizam Md Nasir 和 Md Mujibur Rahman。2021。为冷启动用户推荐获取辅助信息：一项调查。Applied Sciences 11, 20 (2021), 9608。 [2] Gediminas Adomavicius 和 Jingjing Zhang。2016。推荐算法的分类、排名与 Top-K 稳定性。INFORMS Journal on Computing 28, 1 (2016), 129–147。 [3] Charu C Aggarwal 等。2016。推荐系统。第 1 卷。Springer。[4] Vito Walter Anelli、Alejandro Bellogín、Tommaso Di Noia、Dietmar Jannach 和 Claudio Pomo。2022。Top-N推荐算法：探寻最新水平。arXiv 预印本 arXiv:2203.01155 (2022)。[5] Vito Walter Anelli 、 Alejandro Bellogín 、 Tommaso Di Noia 和 Claudio Pomo。2021。重新审视神经协同过滤与矩阵分解之间的比较。载于 ACM推荐系统会议。521–529。 [6] Vito Walter Anelli、Alejandro Bellogín、Antonio Ferrara、Daniele Malitesta、Fe- lice Antonio Merra、Claudio Pomo、Francesco Maria Donini 和 Tommaso Di Noia。2021。Elliot：用于可复现推荐系统评估的全面且严谨的框架。载于国际ACM SIGIR信息检索研究与发展会议论文集。2405–2414。[7] Vito Walter Anelli、Tommaso Di Noia、Eugenio Di Sciascio、Claudio Pomo 和 Azzurra Ragone。2019。交叉验证中超参数的区分能力及其选择方法。载于第 13 届 ACM 推荐系统会议论文集。447–451。 [8] Timothy G Armstrong、Alistair Moffat、William Webber 和 Justin Zobel。2009。未能带来累加收益的改进：1998 年以来的即席检索结果。载于第 18 届 ACM 信息与知识管理会议论文集。601–610。[9] Devansh Arpit、Stanisław Jastrzębski、Nicolas Ballas、David Krueger、Emmanuel Bengio、Maxinder S Kanwal、Tegan Maharaj、Asja Fischer、Aaron Courville、Yoshua Bengio 等。2017。深入探究深度网络中的记忆。载于国际机器学习会议。PMLR，233–242。 [10] James Bennett、Stan Lanning 等。2007。Netflix奖。载于 KDD杯及研讨会论文集，第 2007 卷。纽约，35。 [11] Jiajun Bu、Xin Shen、Bin Xu、Chun Chen、Xiaofei He 和 Deng Cai。2016。通过用户-物品子群改进协同推荐。IEEE知识与数据工程汇刊 28, 9 (2016), 2363–2375。 [12] Joseph A. Konstan、Georg Lausen、Cai-Nicolas Ziegler、Sean M. McNee。2005。通过主题多样化改进推荐列表。载于第 14 届国际万维网会议（WWW '05）论文集。[13] Rocío Cañamares、Pablo Castells 和 Alistair Moffat。2020。推荐系统的离线评估方案。Information Retrieval Journal 23, 4 (2020), 387–410。[14] Satrajit Chatterjee。2018。学习与记忆。载于国际机器学习会议。PMLR，755–763。 [15] Qiwei Chen、Huan Zhao、Wei Li、Pipei Huang 和 Wenwu Ou。2019。用于阿里巴巴电子商务推荐的行为序列变换器。载于首届高维稀疏数据深度学习实践国际研讨会论文集。1–4。[16] Heng-Tze Cheng、Levent Koc、Jeremiah Harmsen、Tal Shaked、Tushar Chandra、Hrishi Aradhye、Glen Anderson、Greg Corrado、Wei Chai、Mustafa Ispir 等。2016。用于推荐系统的宽深度学习。载于首届推荐系统深度学习研讨会论文集。7–10。 [17] Colin Cooper、Sang Hyuk Lee、Tomasz Radzik 和 Yiannis Siantos。2014。推荐系统中的随机游走：精确计算与模拟。载于第 23 届国际万维网会议论文集。811–816。 [18] Paolo Cremonesi、Franca Garzotto 和 Roberto Turrin。2013。以用户为中心与以系统为中心的推荐系统评估。载于 IFIP 人机交互会议。Springer，334–351。 [19] Paolo Cremonesi 和 Dietmar Jannach。2021。推荐系统研究的进展：危机？什么危机？AI Magazine 42, 3 (2021), 43–54。 [20] Paolo Cremonesi、Yehuda Koren 和 Roberto Turrin。2010。推荐算法在 Top-N 推荐任务中的表现。载于第四届 ACM推荐系统会议论文集。39–46。 [21] Maurizio Ferrari Dacrema、Nicolò Felicioni 和 Paolo Cremonesi。2022。具有多个轮播栏的用户界面中的推荐系统离线评估。Frontiers in Big Data 5 (2022)。[22] Aminu Da'u 和 Naomie Salim。2020。基于深度学习方法的推荐系统：系统综述与新方向。Artificial Intelligence Review 53, 4 (2020), 2709–2748。 [23] Gabriel de Souza Pereira Moreira、Felipe Ferreira 和 Adilson Marques da Cunha。2018。使用深度神经网络的新闻会话推荐。载于第三届推荐系统深度学习研讨会论文集。15–23。 [24] Yashar Deldjoo、Alejandro Bellogin 和 Tommaso Di Noia。2021。透过数据特征的视角解释推荐系统的公平性与准确性。Information Processing & Management 58, 5 (2021), 102662。 [25] Mukund Deshpande 和 George Karypis。2004。基于物品的 Top-N 推荐算法。ACM 信息系统汇刊（TOIS）22, 1 (2004), 143–177。

[26] Michael D Ekstrand、Anubrata Das、Robin Burke 和 Fernando Diaz。2012。推荐系统中的公平性。载于《推荐系统手册》。Springer，679–707。[27] Michael D Ekstrand、Michael Ludwig、Joseph A Konstan 和 John T Riedl。2011。重新思考推荐系统研究生态：可复现性、开放性与 LensKit。载于第五届 ACM 推荐系统会议论文集。133–140。[28] Michael D Ekstrand、Mucun Tian、Mohammed R Imran Kazi、Hoda Mehrpouyan 和 Daniel Kluver。2018。探索图书评分与推荐中的作者性别。载于第十二届 ACM 推荐系统会议论文集。242–250。[29] Farzad Eskandanian 和 Bamshad Mobasher。2020。利用稳定匹配优化推荐中准确性与多样性之间的平衡。载于第 28 届 ACM 用户建模、自适应与个性化会议论文集。71–79。[30] Nicolò Felicioni、M Ferrari Dacrema、FB Perez Maurera 和 Paolo Cremonesi。2021。在二维轮播场景中衡量推荐结果的排序质量。载于第 11 届意大利信息检索研讨会（IIR 2021），第 2947 卷。CEUR-WS，1–14。[31] Yufei Feng、Fuyu Lv、Weichen Shen、Menghan Wang、Fei Sun、Yu Zhu 和 Keping Yang。2019。用于点击率预测的深度会话兴趣网络。arXiv 预印本 arXiv:1905.06482（2019）。 [32] Maurizio Ferrari Dacrema、Simone Boglio、Paolo Cremonesi 和 Dietmar J annach。2021。对推荐系统研究的可复现性与进展进行令人担忧的分析。《ACM 信息系统汇刊》（TOIS）39，2（2021），1–49。[33] Maurizio Ferrari Dacrema、Paolo Cremonesi 和 Dietmar Jannach。2019。我们真的取得了很大进展吗？对近期神经推荐方法的令人担忧的分析。载于第十三届 ACM 推荐系统会议论文集。101–109。 [34] Norbert Fuhr。2018。信息检索评估中的一些常见错误及其避免方法。载于《ACM SIGIR论坛》，第 51 卷。ACM，纽约，纽约州，美国，32–41。[35] Chongming Gao、Shijun Li、Wenqiang Lei、Biao Li、Peng Jiang、Jiawei Chen、Xiangnan He、Jiaxin Mao 和 Tat-Seng Chua。2022。KuaiRec：用于推荐系统的全观测数据集。arXiv 预印本 arXiv:2202.10842（2022）。 [36] Ken Goldberg、Theresa Roeder、Dhruv Gupta 和 Chris Perkins。2001。Eigentaste：一种常数时间协同过滤算法。《信息检索》4，2（2001），133–151。[37] Robert M Gray。2011。熵与信息论。Springer Science & Business Media。[38] Huifeng Guo、Ruiming Tang、Yunming Ye、Zhenguo Li 和 Xiuqiang He。2017。DeepFM：一种基于因子分解机的点击率预测神经网络。arXiv 预印本 arXiv:1703.04247（2017）。 [39] Junpeng Guo、Wenxiang Zhang、Weiguo Fan 和 Wenhua Li。2018。利用深度学习融合地理与社会影响，实现个性化兴趣点推荐。《管理信息系统期刊》35，4（2018），1121–1153。 [40] F Maxwell Harper 和 Joseph A Konstan。2015。MovieLens数据集：历史与背景。《ACM交互式智能系统汇刊》5，4（2015），1–19。[41] Xiangnan He、Kuan Deng、Xiang Wang、Yan Li、Yongdong Zhang 和 Meng Wang。2020。LightGCN：简化并增强用于推荐的图卷积网络。载于第 43 届国际 ACM SIGIR 信息检索研究与发展会议论文集。639–648。 [42] Xiangnan He、Lizi Liao、Hanwang Zhang、Liqiang Nie、Xia Hu 和 Tat-Seng Chua。2017。神经协同过滤。载于第 26 届国际万维网会议论文集。173–182。 [43] Binbin Hu、Chuan Shi、Wayne Xin Zhao 和 Philip S Yu。2018。利用基于元路径的背景，通过神经协同注意力模型实现 Top-N 推荐。载于第 24 届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。1531–1540。[44] Neil Hurley 和 Mi Zhang。2011。Top-N 推荐中的新颖性与多样性——分析与评估。《ACM互联网技术汇刊》10，4（2011），1–30。[45] Peter Brusilovsky、Iván Cantador 和 Tsvi Kuflik。2011。第二届推荐系统中的信息异质性与融合研讨会（HetRec 2011）。载于第五届 ACM 推荐系统会议（RecSys'11）论文集。[46] Mohsen Jamali 和 Martin Ester。2009。TrustWalker：一种结合基于信任与基于物品推荐的随机游走模型。载于第 15 届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。397–406。[47] Mohsen Jamali 和 Martin Ester。2009。利用信任网络改进 Top-N 推荐。载于第三届 ACM 推荐系统会议论文集。181–188。[48] Yitong Ji、Aixin Sun、Jie Zhang 和 Chenliang Li。2020。重新审视推荐系统中的流行度基线。载于第 43 届国际 ACM SIGIR 信息检索研究与发展会议论文集。1749–1752。

[49] Ruoming Jin、Dong Li、Jing Gao、Zhi Liu、Li Chen 和 Yang Zhou。2021。迈向更深入地理解用于推荐的线性模型。载于第 27 届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。776–785。 [50] Michael Jugovac、Dietmar Jannach 和 Mozhgan Karimi。2018。Streamingrec：用于基于流的新闻推荐器基准测试的框架。载于第十二届 ACM 推荐系统会议论文集。269–273。 [51] Sergey Kolesnikov、Oleg Lashinin、Michail Pechatov 和 Alexander Kosov。2021。TTRS：Tinkoff交易推荐系统基准。arXiv 预印本 arXiv:2110.05589（2021）。 [52] Joseph A Konstan 和 Gediminas Adomavicius。2013。推动算法推荐系统研究中最佳实践的识别与采纳。载于推荐系统评估中的可复现性与复现国际研讨会论文集。23–28。 [53] Georgia Koutrika。2018。现代推荐系统：从矩阵计算到神经元思维。载于2018年国际数据管理会议论文集。1651–1654。[54] Karl Krauth、Sarah Dean、Alex Zhao、Wenshuo Guo、Mihaela Curmei、Benjamin Recht 和 Michael I Jordan。2020。离线指标能否预测推荐系统的在线性能？arXiv 预印本 arXiv:2011.07931（2020）。 [55] Sara Latifi、Dietmar Jannach 和 Andrés Ferraro。2022。序列推荐：关于Transformer模型、最近邻和采样指标的研究。《信息科学》609（2022），660–678。 [56] Yann LeCun、Yoshua Bengio 和 Geoffrey Hinton。2015。深度学习。《Nature》521，7553（2015），436–444。 [57] Dong Li、Ruoming Jin、Jing Gao 和 Zhi Liu。2020。关于采样式 Top-K 推荐评估。载于第 26 届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。2114–2124。[58] Ruifeng Li、Yin Zhang、Haihan Yu、Xiaojun Wang、Jiangqin Wu 和 Baogang Wei。2011。一种使用 GPU 的社交网络感知 Top-N 推荐系统。载于第 11 届年度国际 ACM/IEEE 数字图书馆联合会议论文集。287–296。[59] Yunqi Li、Hanxiong Chen、Zuohui Fu、Yingqiang Ge 和 Yongfeng Zhang。2021。推荐中的面向用户的公平性。载于2021年 Web会议论文集。624–632。[60] Dawen Liang、Rahul G Krishnan、Matthew D Hoffman 和 Tony Jebara。2018。用于协同过滤的变分自编码器。载于2018年万维网大会论文集。689–698。 [61] Malte Ludewig 和 Dietmar Jannach。2018。基于会话的推荐算法评估。《用户建模与用户适应性交互》28，4（2018），331–390。[62] Malte Ludewig、Noemi Mauro、Sara Latifi 和 Dietmar Jannach。2019。神经方法与非神经方法在基于会话的推荐中的性能比较。载于第十三届 ACM 推荐系统会议论文集。462–466。 [63] Benjamin M Marlin。2003。协同过滤中的用户评分档案建模。《神经信息处理系统进展》16（2003）。 [64] Benjamin M Marlin 和 Richard S Zemel。2009。利用非随机缺失数据进行协同预测与排名。载于第三届 ACM 推荐系统会议论文集。5–12。 [65] Paolo Massa 和 Paolo Avesani。2007。信任感知推荐系统。载于2007年 ACM 推荐系统会议论文集。17–24。 [66] Julian McAuley、Christopher Targett、Qinfeng Shi 和 Anton Van Den Hengel。2015。基于风格与替代品图像的推荐。载于第 38 届国际 ACM SIGIR 信息检索研究与发展会议论文集。43–52。 [67] Zaiqiao Meng、Richard McCreadie、Craig Macdonald 和 Iadh Ounis。2020。探索推荐模型评估中的数据划分策略。载于第十四届 ACM 推荐系统会议。681–686。 [68] Xia Ning 和 George Karypis。2011。SLIM：用于 Top-N 推荐系统的稀疏线性方法。载于2011年 IEEE 第 11 届国际数据挖掘会议。IEEE，497–506。 [69] Bibek Paudel、Fabian Christoffel、Chris Newell 和 Abraham Bernstein。2016。面向交互式应用的可更新、准确、多样且可扩展的推荐。《ACM交互式智能系统汇刊》7，1（2016），1–34。[70] Fernando Benjamín Pérez Maurera、Maurizio Ferrari Dacrema 和 Paolo Cremonesi。2022。生成对抗网络在协同过滤中的评估研究。载于欧洲信息检索会议。Springer，671–685。[71] Lijing Qin 和 Xiaoyan Zhu。2013。通过熵正则化项提升推荐的多样性。载于第二十三届国际人工智能联合会议。Citeseer。 [72] Hossein A Rahmani、Yashar Deldjoo、Ali Tourani 和 Mohammadmehdi Naghiaei。2022。兴趣点推荐中活跃用户的不公平性与流行度偏差。arXiv 预印本 arXiv:2202.13307（2022）。 [73] Chamsi Abu Quba Rana、Hassas Salima、Fayyad Usama 和 Chamsi Hammam。2014。从“冷启动”到“热启动”的推荐系统。载于2014年 IEEE 第 23 届国际 WETICE会议。IEEE，290–292。

[74] Steffen Rendle、Li Zhang 和 Yehuda Koren。2019。基线评估的难点：推荐系统研究。arXiv 预印本 arXiv:1905.01395（2019）。[75] Noveen Sachdeva、Carole-Jean Wu 和 Julian McAuley。2022。关于协同过滤数据集的采样。arXiv 预印本 arXiv:2201.04768（2022）。[76] Alan Said 和 Alejandro Bellogín。2014。推荐系统评估比较：推荐框架的基准测试。载于《第八届 ACM 推荐系统会议论文集》，129–136。[77] J Ben Schafer、Joseph A Konstan 和 John Riedl。2001。电子商务推荐应用。《数据挖掘与知识发现》5（2001），115–153。[78] Suvash Sedhain、Aditya Menon、Scott Sanner 和 Darius Braziunas。2016。线性模型在单类协同过滤中的有效性。载于《AAAI 人工智能会议论文集》，第 30 卷。[79] Yifei Shen、Yongji Wu、Yao Zhang、Caihua Shan、Jun Zhang、B Khaled Letaief 和 Dongsheng Li。2021。图卷积对推荐有多强大？载于《第 30 届 ACM 国际信息与知识管理会议论文集》，1619–1629。[80] Harald Steck。2019。适用于稀疏数据的浅层自编码器。《万维网大会论文集》，3251–3257。[81] Ke Sun、Tieyun Qian、Tong Chen、Yile Liang、Quoc Viet Hung Nguyen 和 Hongzhi Yin。2020。下一步去哪里：对兴趣点推荐中用户长期与短期偏好的建模。载于《AAAI 人工智能会议论文集》，第 34 卷，214–221。[82] Zhu Sun、Hui Fang、Jie Yang、Xinghua Qu、Hongyang Liu、Di Yu、Yew-Soon Ong 和 Jie Zhang。2022。DaisyRec 2.0：严格评估推荐的基准测试。《IEEE 模式分析与机器智能汇刊》（2022）。[83] Zhu Sun、Di Yu、Hui Fang、Jie Yang、Xinghua Qu、Jie Zhang 和 Cong Geng。2020。我们的评估是否严谨？面向可复现评估与公平比较的推荐基准测试。载于《第十四届 ACM 推荐系统会议论文集》，23–32。[84] Daniel Valcarce、Alejandro Bellogín、Javier Parapar 和 Pablo Castells。2018。信息检索指标用于 Top-N 推荐的鲁棒性与判别力。载于《第十二届 ACM 推荐系统会议论文集》，260–268。[85] Hongwei Wang、Fuzheng Zhang、Xing Xie 和 Minyi Guo。2018。DKN：用于新闻推荐的深度知识感知网络。载于《2018 年万维网大会论文集》，1835–1844。[86] Jizhe Wang、Pipei Huang、Huan Zhao、Zhibo Zhang、Binqiang Zhao 和 Dik Lun Lee。2018。用于阿里巴巴电子商务推荐的十亿级商品嵌入。载于《第 24 届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集》，839–848。[87] Wenjie Wang、Fuli Feng、Xiangnan He、Liqiang Nie 和 Tat-Seng Chua。2021。为推荐中的隐式反馈去噪。载于《第 14 届 ACM 国际网络搜索与数据挖掘会议论文集》，373–381。[88] Colin Wilkie 和 Leif Azzopardi。2014。最佳且最公平：检索系统偏差的实证分析。载于《信息检索进展：第 36 届欧洲信息检索研究会议 ECIR 2014》，阿姆斯特丹，荷兰，2014 年 4 月，论文集。Springer，13–25。[89] Bin Xu、Jiajun Bu、Chun Chen 和 Deng Cai。2012。探索通过用户-物品子群改进协同推荐系统。载于《第 21 届万维网国际会议论文集》，21–30。[90] Yuan Yao 和 F Maxwell Harper。2018。相似性评判：一项以用户为中心的相关物品推荐研究。载于《第十二届 ACM 推荐系统会议论文集》，288–296。[91] Quan Yuan、Gao Cong、Zongyang Ma、Aixin Sun 和 Nadia Magnenat Thalmann。2013。时间感知兴趣点推荐。载于《第 36 届 ACM SIGIR 信息检索研究与发展国际会议论文集》，363–372。[92] Chiyuan Zhang、Samy Bengio、Moritz Hardt、Benjamin Recht 和 Oriol Vinyals。2017。理解深度学习需要重新思考泛化（2016）。arXiv 预印本 arXiv:1611.03530（2017）。[93] Mi Zhang 和 Neil Hurley。2008。避免单调：提升推荐列表的多样性。载于《2008 年 ACM 推荐系统会议论文集》，123–130。[94] Wayne Xin Zhao、Zihan Lin、Zhichao Feng、Pengfei Wang 和 Ji-Rong Wen。2022。Top-N 推荐算法适当离线评估的再探讨。《ACM 信息系统汇刊》（TOIS）（2022）。[95] Wayne Xin Zhao、Shanlei Mu、Yupeng Hou、Zihan Lin、Yushuo Chen、Xingyu Pan、Kaiyuan Li、Yujie Lu、Hui Wang、Changxin Tian 等。2021。Recbole：迈向统一、全面且高效的推荐算法框架。载于《第 30 届 ACM 国际信息与知识管理会议论文集》，4653–4664。[96] Guanjie Zheng 、Fuzheng Zhang、Zihan Zheng、Yang Xiang、Nicholas Jing Yuan、Xing Xie 和 Zhenhui Li。2018。DRN：用于新闻推荐的深度强化学习框架。载于《2018 年万维网大会论文集》，167–176。