使用大型预训练语言模型进行零样本下一物品推荐

Lei Wang Ee-Peng Lim*
lei.wang.2019
@phdcs.smu.edu.sg
eplim@smu.edu.sg 新加坡管理大学
新加坡

arXiv:2304.03153v1 [cs.IR] 6 Apr 2023

摘要

大型语言模型（大语言模型）在各种自然语言处理（NLP）任务中展现出令人瞩目的零样本表现，证明了它们无需训练示例即可进行推理的能力。尽管取得了这些成功，目前尚无研究探索大语言模型在零样本设置下执行下一项目推荐的潜力。我们发现，要使大语言模型有效地充当推荐系统，必须应对两大挑战。首先，推荐空间对于大语言模型而言可能极其庞大，而且大语言模型不了解目标用户过去交互过的物品及其偏好。为弥补这一不足，我们提出了一种名为**零样本下一物品推荐（NIR）**提示的提示策略，引导大语言模型进行下一项目推荐。具体而言，基于 NIR 的策略使用外部模块，通过用户筛选或物品筛选生成候选物品。我们的策略采用三步提示，引导 GPT-3 执行若干子任务，以捕捉用户偏好、挑选具有代表性的已观看电影，并推荐包含 10 部电影的排序列表。我们在 MovieLens 100K数据集上使用 GPT-3 评估了所提出的方法，结果表明，该方法取得了出色的零样本表现，甚至超过了一些在完整训练数据集上训练的强大序列推荐模型。这些令人鼓舞的结果凸显了将大语言模型用作推荐系统的广阔研究机遇。代码见
https://github.com/AGI-Edgerunners/LLM-Next-Item-Rec。

CCS 概念

• **信息系统 →推荐系统**.

关键词

下一项目推荐、大型语言模型、零样本学习、提示

*通讯作者。

允许为个人或课堂使用而免费制作本作品的数字或纸质副本（无论是全部还是部分），前提是副本不得用于或分发以获取利润或商业利益，且副本首页须注明本声明和完整引文。必须尊重本作品中由 ACM 以外的其他方持有版权的部分。注明出处后，可以进行摘要。若要以其他方式复制、再版、发布到服务器或向列表重新分发，须事先获得特定许可和/或支付费用。请通过 permissions@acm.org 申请许可。
*Conference'17, 2017*年7月, 美国华盛顿特区© 2023 Association for Computing Machinery。ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

ACM参考格式：
Lei Wang 和 Ee-Peng Lim*。 2023。使用大型预训练语言模型进行零样本下一物品推荐。载于*ACM Conference* 论文集（*Conference'17*）。ACM，纽约州纽约市，5页。https://doi.org/10.1145/ nnnnnnn.nnnnnnn

1 引言

大型语言模型（大语言模型） [1, 3, 25]，（例如 GPT-3 [1], ）在各种自然语言处理（NLP）任务中取得了令人瞩目的成果。然而，大语言模型规模也非常庞大，通常只能通过某些 API 服务访问。因此，它们无法像早期的预训练语言模型（PTM） [5, 15]那样进行微调。许多研究表明，在零样本设置下，即不提供任何示范样例或进行进一步训练 [1, 3]，大语言模型能够通过特定任务的提示解决许多已知的 NLP 问题。尽管如此，在零样本设置下使用大语言模型进行下一项目推荐仍处于研究起步阶段。

我们用图 1 展示自然语言处理推理任务与推荐任务之间的差异。在这项自然语言处理推理任务中，向 GPT-3 [12] （也是本文默认使用的大语言模型）提供一个提示，其中包含一个问题，随后由模型生成答案文本（例如"9"）。与这类可以直接依赖大语言模型内置文本知识的自然语言处理任务不同，推荐任务要求大语言模型了解目标用户过去与物品的交互、待推荐物品的全集，以及选择推荐物品的恰当方法。鉴于大语言模型并非专门针对推荐任务进行训练，直接用它们执行推荐预计效果不佳 [26]。此外，只有在对待推荐物品具备一定背景知识时，大语言模型才能为推荐提供帮助。对于专有领域中的物品推荐，目前尚不清楚大语言模型能发挥多大作用。

因此，在我们的研究中，我们假设待推荐的物品应出现在大语言模型的训练数据中。这类物品包括电影、歌曲、小说、在线游戏等。为了进行说明和评估，我们聚焦于下一部电影推荐任务。我们还选择 GPT-3 作为大语言模型，因为它广受欢迎且易于使用。如图 1(b) 所示，我们展示了一种简单的提示策略：将用户之前看过的电影直接纳入文本提示中，即"根据我看过的电影，包括……"，然后提出问题："你能给我推荐 10 部电影吗？"尽管这种提示策略可以让 GPT-3 充当电影推荐器，但其推荐准确率……

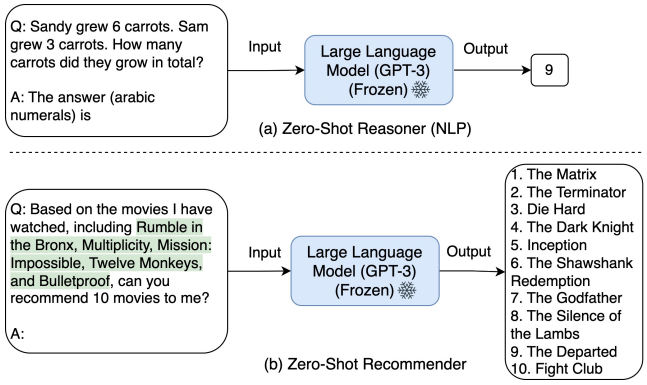


图 1：GPT-3 在零样本提示下针对自然语言处理任务和推荐任务的示例输入与输出。

可能会因极其庞大的推荐空间和不充分的用户偏好建模而表现不佳。

因此，本文提出了一种有原则的下一项目推荐方法，称为**零样本下一项目推荐（NIR）提示**。该方法采用三步提示策略，在零样本设置下显著优于简单提示。零样本下一项目推荐采用三管齐下的方法来提高推荐准确率。首先，通过运用以往下一项目推荐研究中广为人知的用户筛选和项目筛选技术，为目标用户生成候选电影集，将推荐范围限制在 MovieLens [7] 数据集内。其次，零样本下一项目推荐提示策略通过多步提示 GPT-3 来捕捉用户偏好（步骤 1）、从用户此前看过的电影中选出代表性电影（步骤 2），并推荐由 10 部电影组成的排序列表（步骤 3）。最后，零样本下一项目推荐在步骤 3 中引入格式化技术，以便从 GPT-3 生成的回答文本中提取电影项目（即"已观看电影： <-候选电影->"）。我们使用 MovieLens 100K 和 GPT-3 引擎 text-davinci-003 对该方法进行评估。实验结果表明，零样本下一项目推荐提示在零样本设置下取得了良好的推荐准确率，与使用大型数据集训练的其他下一项目推荐方法表现相当。

2 相关工作

下一项目推荐是一项重要且经过充分研究的问题。早期研究提出了马尔可夫链，用于建模项目之间的低阶关系，以实现下一项目推荐 [8, 16]。随着神经网络模型的发展，深度神经网络 [2, 9–11, 18–20] 已被用于建模序列模式，从而提高了推荐准确率。近期研究还探索了利用数据增强和对比学习来增强用户和项目的表示，进而进一步提升推荐性能 [14, 21, 23, 24, 27]。然而，上述所有方法都需要使用用户的历史项目交互进行模型训练。换言之，它们无法在零样本设置下进行推荐。据我们所知，尚未有人.....

据我们所知，关于零样本推荐的研究非常少，而且目前尚不清楚大语言模型能否成为优秀的推荐器。

在基于大语言模型的推荐早期研究中，[4, 6, 13, 17, 22, 26]，[26] 2 [15]

Zhang 等人提出使用 GPT 或 BERT [5] 作为骨干推荐模型，根据目标用户此前观看过的五部电影进行下一部电影预测。然而，庞大的推荐空间和不足的用户偏好建模使其表现不佳。鉴于 GPT-3 [1], OPT [25], 和 PaLM [3] 等较新的大语言模型在各种自然语言处理任务中取得了显著提升的结果，我们选择 GPT-3 作为大语言模型，以开发更有效的零样本推荐方法。我们提出的零样本 NIR 提示策略并非从头设计提示策略，而是结合用户和项目筛选方法来生成候选电影集，并设计了一个三步提示方法。通过这种方式，它模拟了众所周知的推荐技术，从而实现更准确的零样本推荐。

3 零样本下一项目推荐提示策略

3.1 概述

零样本 NIR 提示是一种多步提示策略，使GPT-3能够在零样本设置下充当下一项目推荐器。图 2 展示了我们的零样本 NIR 提示策略的完整过程。它由三个组件组成：

• **候选集构建：** 该组件使用用户筛选或项目筛选，为每个目标用户创建候选集，从而缩小推荐空间。随后，这些候选电影被用于构建三步式 GPT-3 提示。

• **GPT-3 三步提示：** 该组件涉及对应三个子任务的三个指令提示。在第一个子任务（用户偏好子任务）中，我们设计了一个用户偏好提示，以探测 GPT-3基于目标用户此前互动过的电影来总结用户的偏好。在第二个子任务（代表性电影子任务）中，我们创建了一个提示，将用户偏好提示与提示回答以及触发指令相结合，请求GPT-3 按偏好降序选择代表性电影。在第三个子任务中，我们整合代表性电影提示、其回答和一个问题，创建推荐提示，以引导GPT-3 从候选电影集中推荐10部与代表性电影相似的电影。预期结果格式如下："一部已观看的电影： <- 一部候选电影 ->"。

• **答案提取：** 该组件采用简单的基于规则的提取方法，从 GPT-3 三步提示生成的文本结果中提取推荐项目。提取出的推荐电影结果可用于下游应用或性能评估。

3.2 候选集构建

如第1节所述，极其庞大的推荐空间给基于大语言模型的推荐带来了重大挑战。无约束的推荐空间可能会使实际

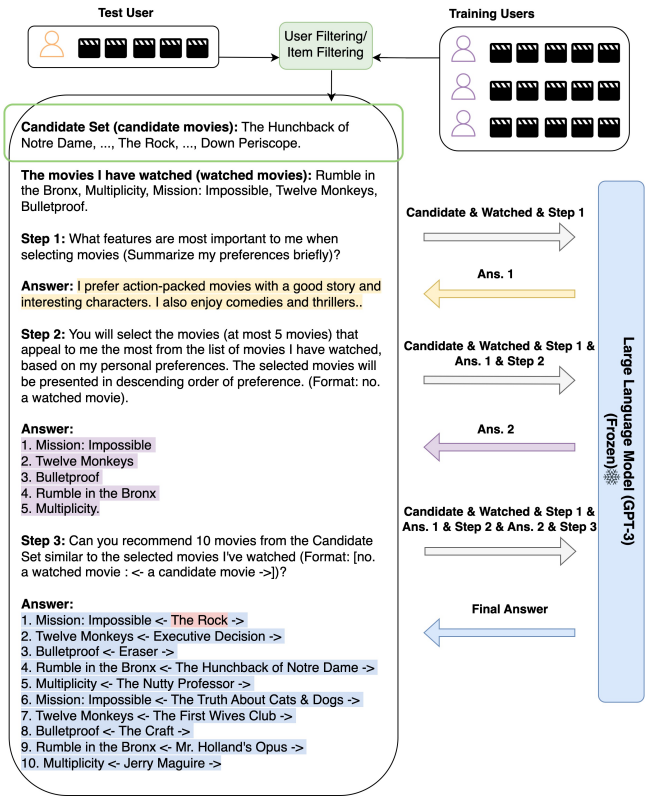


图 2：零样本下一项目推荐提示。真实电影（即 TheRock）已用红色标出。

推荐结果的使用和评估。另一方面，将所有项目输入大语言模型，并期待其了解项目全集，是不可行的。即使是为了评估性能，MovieLens100K数据集也包含1,683部电影，这个数量仍然太大，无法放入提示中。

因此，在零样本下一项目推荐提示中，我们提出先采用一种有依据的方法，为每个目标用户构建候选电影集。候选电影应满足两个标准：(a) 它们应与用户更加相关，而非随机选取；(b) 候选电影的数量应较少，以便放入提示中。为满足这些标准，我们采用两种用于确定候选电影的公认原则，即用户筛选和项目筛选。

用户筛选。该原则假设候选电影也应受到与目标用户相似的其他用户的喜爱。因此，我们首先用一个多热向量表示每位用户看过的电影。随后，通过计算目标用户向量与其他用户向量之间的余弦相似度，找出与目标用户相似的用户。接下来，我们选取 m 位最相似的用户，并通过从相似用户互动过的电影中挑选最受欢迎的电影，构建大小为 s 的候选电影集。

项目筛选。与用户筛选类似，我们根据与其交互过的用户，为每部电影表示一个多热向量。利用两部电影之间的余弦相似度，我们为目标用户的交互历史中的每部电影选择 n 部最相似的电影。然后，我们根据这些相似电影在目标用户交互历史中的"受欢迎程度"，生成一个大小为 s 的候选集。

候选集构建可以由外部模块预先完成。随后，如图 2 所示，我们使用"候选集（候选电影）："这一句式，将候选电影纳入后续的推荐提示中。在候选集之后，提示还会列出目标用户此前互动过的电影，开头为："我看过的电影（看过的电影）："。

3.3 三步提示

图 1(b) 显示，简单提示方法忽略了候选电影、用户偏好以及最能反映用户口味的已互动电影。相比之下，我们提出的三步提示方法通过三轮提示完成三个子任务：捕捉目标用户的偏好、根据用户偏好对其此前互动过的电影进行排序，以及从候选集中推荐 10 部相似电影。

步骤 1：用户偏好提示。为了捕捉用户偏好，我们在第一个提示中加入了这样一句话："步骤 1：选择电影时，哪些特征对我最重要（请简要概括我的偏好）？"如图 2 所示，GPT-3 返回的答案概括了目标用户的偏好（以黄色高亮显示）。

步骤2：代表性电影选择提示。作为第二步，该提示包含附加了步骤 1 答案的先前提示文本。随后，它加入以下指令："步骤 2：你将选择.....最吸引我的电影.....并按照偏好降序排列（.....）"，以确定最能反映目标用户品味的先前互动过的电影。图 2 展示了 GPT-3 的答案，其中以紫色突出显示。

步骤 3：推荐提示。同样，此提示包括附加了第 2 步答案的前文。随后，它包含指令"步骤3：你能从 CandidateSet中推荐 10 部与.....相似的电影吗？"。此提示明确指示 GPT-3 从候选集中生成 10 部推荐电影，如蓝色所突出显示。

3.4 答案提取

我们在第三个提示中加入了"（格式：... <-一部候选电影 ->]）"这一提示，以生成所需格式的答案，便于提取。我们的研究表明，这样的提示效果非常好。

4 实验

4.1 实验设置

数据集。 所提出的提示方法在广泛使用的电影推荐数据集 MovieLens 100K [7], 上进行了评估，该数据集包含 944 名用户和 1,683 部电影。

基线。 我们将提出的 NIR 提示策略与两类基线进行比较：强大的下一项目推荐基线 和 零样本基线。前者包括 **POP** （一种基于流行度的模型）、**FPMC** [16] （一种结合矩阵分解和马尔可夫链的方法）、**GRU4** [9] （一种基于 GRU 的序列

表1：MovieLen 100K 上的主要结果比较。

设置	方法	MovieLens 100K	
		HR@10	NDCG@10
全量训练	POP	0.0519	0.0216
	FPMC	0.1018	0.0463
	GRU4Rec	0.1230	0.0559
	SASRec	0.1241	0.0573
	CL4SRec	0.1273	0.0617
零样本	简单提示	0.0297	0.0097
	CS-Combine-IF	0.0805	0.0352
	CS-Combine-UF	0.0954	0.0457
	NIR-Single-IF	0.0975	0.0501
	NIR-Single-UF	0.1135	0.0529
	NIR-Multi-IF	0.1028	0.0505
	NIR-Multi-UF	0.1187	0.0546

推荐模型）、**SASRec** [11]（一种采用自注意力机制的序列推荐模型），以及 **CL4SRec** [23]（一种基于对比学习的序列推荐模型）。由于这些强基线具备完整训练模型的优势，预计其表现将优于零样本方法。零样本基线包括 **零样本简单提示**、**CS-Random-IF**（从基于项目筛选的候选集中随机选取 10 部电影），以及 **CS-Random-UF**（从基于 UF 的候选集中随机选取 10 部电影）。我们采用四种变体实现 NIR 提示策略。**NIR-Combine-IF/NIR-Combine-UF**将 3 个步骤合并为一个提示，不保留中间答案。我们仅向 GPT-3 提示一次，使其从基于 IF/UF 的候选集中生成 10 部推荐电影。**NIR-Multi-IF/NIR-Multi-UF**使用三个独立的提示逐步引导 GPT-3，并将中间答案纳入后续提示（如图 2 所示），同时使用基于 IF/UF 的候选集。**实现**。在实验中，我们采用公开的 GPT-3 text-davinci-003（175B）作为骨干语言模型；它是最受欢迎且提供公开 API 的大语言模型之一[1]。为确保输出一致，我们将温度参数设为 0。评估时，我们采用与 CL4SRec 相同的评估指标，并报告所有方法的 HR@10 和 NDCG@10。

4.2 主要结果

如表1 所示，我们基于 NIR的方法（NIR-Single-IF、NIR-Single-UF、NIR-Multi-IF、NIR-Multi-UF）以较大优势超越了 POP基线。有趣的是，NIR-Single-UF、NIR-Multi-IF 和 NIR-Multi-UF 始终优于 FPMC（一种完全训练的方法）。与非常强大的序列推荐模型（即GRU4Rec、SASRec和CL4SRec）相比，这三种基于 NIR 的方法的表现仍略逊一筹，但具有竞争力，这表明采用适当提示策略的大语言模型可以成为相当不错的零样本推荐器。

在零样本设置中，CS-Random-UF(IF)相较于Simple Prompting 的优越表现表明，候选集不仅缩小了推荐空间，还提升了性能。我们提出的基于 NIR 的提示方法始终表现更优

[1]https://beta.openai.com/docs/models/gpt-3

表 2：在 MovieLens100K 上，对所提出的提示方法中不同组件的影响进行的消融实验。

候选集	用户偏好	代表性电影	HR@10
–	–	–	0.0297
✓	–	–	0.1071
✓	✓	–	0.1136
✓	–	✓	0.1082
✓	✓	✓	0.1187

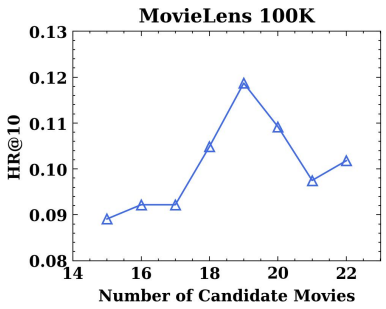


图 3：不同候选集规模下的结果。

Simple 提示和 CS-Random-IF/UF 的表现表明，在提示过程中融入用户偏好、代表性电影选择和格式设置技巧，能够让 GPT-3 给出更好的推荐。由于 Multi-IF(UF) 优于 Combine-IF(UF)，我们知道，将中间答案纳入后续提示的独立提示能够带来更准确的推荐。最后，基于 UF 的 HIR 提示始终优于基于 IF 的提示，说明 UF 生成的候选集优于 IF。

4.3 详细分析

提示各组成部分的效果。我们现在对 NIP-Multi-UF 进行消融研究，以评估提示策略中不同组成部分的贡献。表 2 显示，所有提示组成部分都有助于提升推荐性能。采用候选集的简单提示方法（HR@10=0.1071）优于不采用候选集的方法（HR@10=0.0297）。在提示中加入用户偏好或选择具有代表性的电影，都能提升性能，这表明针对特定任务的指令可以引导 GPT-3 更好地完成任务。

候选集大小的影响。我们通过将候选电影数量从 15 调整到 22，同时保持其他参数不变，来研究候选集大小的影响。图 3 的结果表明，推荐性能对候选集大小十分敏感。候选电影数量为 19 时效果最佳，而候选集较小或较大都会导致性能下降。一种可能的解释是，候选集较小会限制性能上限，而候选集较大则会增加使用 GPT-3 做出准确推荐的难度。

5 结论

本文提出了一种名为零样本下一物品推荐（NIR）的三步提示策略，使 GPT-3 无需进一步训练即可进行下一部电影推荐。我们在电影推荐数据集上评估了该方法，并展示了其出色的零样本性能。研究结果凸显了大语言模型在零样本推荐中的潜力，并呼吁进一步探索大语言模型在推荐任务中的应用。这项工作可以从多个方向拓展，包括其他领域的推荐以及少样本设置。

参考文献

[1] Tom B. Brown、Benjamin Mann、Nick Ryder、Melanie Subbiah、Jared Kaplan、Prafulla Dhariwal、Arvind Neelakantan、Pranav Shyam、Girish Sastry、Amanda Askell、Sandhini Agarwal、Ariel Herbert-Voss、Gretchen Krueger、Tom Henighan、Rewon Child、Aditya Ramesh、Daniel M. Ziegler、Jeffrey Wu、Clemens Winter、Christopher Hesse、Mark Chen、Eric Sigler、Mateusz Litwin、Scott Gray、Benjamin Chess、Jack Clark、Christopher Berner、Sam McCandlish、Alec Radford、Ilya Sutskever 和 Dario Amodei。2020。《语言模型是少样本学习者》。载于神经信息处理系统进展 *33*：神经信息处理系统年会 *2020，NeurIPS 2020，2020* 年 *12* 月 *6—12* 日，线上举行，Hugo Larochelle、Marc'Aurelio Ranzato、Raia Hadsell、Maria-Florina Balcan 和 Hsuan-Tien Lin（编）。https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html[2] Jianxin Chang、Chen Gao、Yu Zheng、Yiqun Hui、Yanan Niu、Yang Song、Depeng Jin 和 Yong Li。2021。基于图神经网络的序列推荐。载于第 *44* 届 *ACM SIGIR* 信息检索研究与发展国际会议论文集。378–387。[3] Aakanksha Chowdhery、Sharan Narang、Jacob Devlin、Maarten Bosma、Gaurav Mishra、Adam Roberts、Paul Barham、Hyung Won Chung、Charles Sutton、Se- bastian Gehrmann 等。2022。PaLM：利用路径扩展语言建模规模。*arXiv* 预印本abs/2204 02311 2022 2204 02311。（）。[4] https://arxiv.org/abs/ 。Zeyu Cui、Jianxin Ma、Chang Zhou、Jingren Zhou 和 Hongxia Yang。2022。M6- Rec：生成式预训练语言模型是开放式推荐系统。*arXiv* 预印本 *arXiv:2205.08084*（2022）。[5] Jacob Devlin、Ming-Wei Chang、Kenton Lee 和 Kristina Toutanova。2018。BERT：用于语言理解的深度双向 Transformer 预训练。*arXiv* 预印本 *arXiv:1810.04805*（2018）。[6] Shijie Geng、Shuchang Liu、Zuohui Fu、Yingqiang Ge 和 Yongfeng Zhang。2022。将推荐视为语言处理（RLP）：统一的预训练、个性化提示与预测范式（P5）。*arXiv* 预印本 *arXiv:2203.13366*（2022）。[7] 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（*TIIS*）5，4（2015），1–19。[8] Ruining He 和 Julian McAuley。2016。将相似度模型与马尔可夫链结合，用于稀疏序列推荐。载于*2016* 年 *IEEE* 第 *16* 届国际数据挖掘会议（*ICDM*）。IEEE，191–200。[9] Balázs Hidasi、Alexandros Karatzoglou、Linas Baltrunas 和 Domonkos Tikk。2015。基于循环神经网络的会话推荐。*arXiv* 预印本 *arXiv:1511.06939*（2015）。[10] Jin Huang、Wayne Xin Zhao、Hongjian Dou、Ji-Rong Wen 和 Edward Y Chang。2018。利用知识增强记忆网络改进序列推荐。载于第 *41* 届 *ACM SIGIR* 信息检索研究与发展国际会议。505–514。[11] Wang-Cheng Kang 和 Julian McAuley。2018。自注意力序列推荐。载于*2018* 年 *IEEE* 国际数据挖掘会议（*ICDM*）。IEEE，197–206。[12] Takeshi Kojima、Shixiang Shane Gu、Machel Reid、Yutaka Matsuo 和 Yusuke Iwasawa。2022。大型语言模型是零样本推理器。*arXiv* 预印本 *arXiv:2205.11916*（2022）。[13] Lei Li、Yongfeng Zhang 和 Li Chen。2022。面向可解释推荐的个性化提示学习。*arXiv* 预印本 *arXiv:2202.07371*（2022）。[14] Zhiwei Liu、Yongjun Chen、Jia Li、Man Luo、S Yu Philip 和 Caiming Xiong。2021。通过模型增强进行序列推荐的自监督学习。（2021）。[15] Alec Radford、Jeff Wu、Rewon Child、David Luan、Dario Amodei 和 Ilya Sutskever。2019。语言模型是无监督的多任务学习者。（2019）。[16] Steffen Rendle、Christoph Freudenthaler 和 Lars Schmidt-Thieme。2010。将个性化马尔可夫链分解用于下一篮子推荐。载于第 *19* 届万维网国际会议论文集。811–820。[17] Damien Sileo、Wout Vossen 和 Robbe Raymaekers。2022。将零样本推荐视为语言建模。载于欧洲信息检索会议。Springer，223–230。[18] Fei Sun、Jun Liu、Jian Wu、Changhua Pei、Xiao Lin、Wenwu Ou 和 Peng Jiang。2019。BERT4Rec：基于 Transformer 双向编码器表示的序列推荐。载于第 *28* 届 *ACM* 信息与知识管理国际会议论文集。1441–1450。[19] Jiaxi Tang 和 Ke Wang。2018。通过卷积序列嵌入实现个性化 top-n 序列推荐。载于第十一届 *ACM* 网络搜索与数据挖掘国际会议论文集。565–573。[20] Jianling Wang、Kaize Ding、Liangjie Hong、Huan Liu 和 James Caverlee。2020。基于序列超图的下一项目推荐。载于第 *43* 届 *ACM SIGIR* 信息检索研究与发展国际会议论文集。1101–1110。[21] Jiancan Wu、Xiang Wang、Fuli Feng、Xiangnan He、Liang Chen、Jianxun Lian 和 Xing Xie。2021。用于推荐的自监督图学习。载于第 *44* 届 *ACM SIGIR* 信息检索研究与发展国际会议论文集。726–735。[22] Yiqing Wu、Ruobing Xie、Yongchun Zhu、Fuzhen Zhuang、Xu Zhang、Leyu Lin 和 Qing He。2022。序列推荐的个性化提示。*arXiv* 预印本 *arXiv:2205.09666*（2022）。[23] Xu Xie、Fei Sun、Zhaoyang Liu、Shiwen Wu、Jinyang Gao、Bolin Ding 和 Bin Cui。2020。序列推荐的对比学习。*arXiv* 预印本 *arXiv:2010.14395*（2020）。[24] Tiansheng Yao、Xinyang Yi、Derek Zhiyuan Cheng、Felix Yu、Ting Chen、Aditya Menon、Lichan Hong、Ed H Chi、Steve Tjoa、Jieqi Kang 等。2021。大规模物品推荐的自监督学习。载于第 *30* 届 *ACM* 信息与知识管理国际会议论文集。4321–4330。[25] Susan Zhang、Stephen Roller、Naman Goyal、Mikel Artetxe、Moya Chen、Shuohui Chen、Christopher Dewan、Mona T.Diab、Xian Li、Xi VictoriaLin、Todor Mihaylov、Myle Ott、Sam Shleifer、Kurt Shuster、Daniel Simig、Punit Singh Koura、Anjali Sridhar、Tianlu Wang 和 Luke Zettlemoyer。2022。OPT：开放式预训练 Transformer 语言模型。*CoRR* abs/2205.01068（2022）。arXiv:2205.01068 https://doi.org/10.48550/arXiv.2205.01068[26] Yuhui Zhang、Hao Ding、Zeren Shui、Yifei Ma、James Zou、Anoop Deoras 和 Hao Wang。2021。作为推荐系统的语言模型：评估与局限。载于我（仍然）不敢相信它没有更好！*NeurIPS 2021* 研讨会。[27] Kun Zhou、Hui Wang、Wayne Xin Zhao、Yutao Zhu、Sirui Wang、Fuzheng Zhang、Zhongyuan Wang 和 Ji-Rong Wen。2020。S3-rec：通过互信息最大化实现序列推荐的自监督学习。载于第 *29* 届 *ACM* 信息与知识管理国际会议论文集。1893–1902。