《LLaRA: Large Language-Recommendation Assistant》复现报告

摘要

本文旨在复现《LLaRA: Large Language-Recommendation Assistant》论文中的实验和模型框架，重点验证 LLaRA 在顺序推荐任务中的表现。LLaRA 框架创新性地结合了传统推荐模型的行为模式和大语言模型（LLM）的语义理解能力，通过混合提示方法（Hybrid Prompting）将用户行为信息与项目的文本特征结合，以提升推荐准确度。此外，LLaRA 采用课程学习（Curriculum Learning）策略，使模型能够逐步适应任务的复杂性，进一步提高训练效率。

在复现过程中，我们使用了 MovieLens、Steam 和 LastFM 三个标准数据集进行实验。实验结果显示，LLaRA 在 HitRatio@1 等评估指标上优于传统的顺序推荐模型（如 SASRec、GRU4Rec、Caser）以及基于 LLM 的其他方法，验证了其项目表示方法的有效性。尤其是 LLaRA 能够有效地融合行为标记和文本标记，提升了模型的推荐准确性和生成推荐的有效性。

本报告总结了 LLaRA 的复现过程，并对比了不同方法的性能，为顺序推荐领域中大语言模型与传统推荐模型的结合提供了有力的实验支持。

关键词：大语言模型；序列推荐

1 引言

推荐系统作为人工智能与数据科学的重要应用领域，已经广泛应用于电子商务、流媒体服务、社交平台等场景。在推荐任务中，顺序推荐（Sequential Recommendation）因其基于用户历史行为序列的预测能力而受到特别关注。然而，传统顺序推荐模型主要依赖用户的行为模式，忽略了项目本身的语义信息和世界知识。这种局限性使得推荐性能在面对多样化场景时受到挑战。

近年来，大语言模型（LLM）的快速发展为推荐系统引入了新的思路。LLM 具备强大的语言理解和推理能力，可以通过处理项目的文本特征（如标题、描述等）来补充传统推荐模型中的语义信息缺失问题。因此，如何将 LLM 的能力与传统推荐模型有效结合，成为推荐系统研究中的一个前沿方向。

本文的主要目的是复现并分析《LLaRA: Large Language-Recommendation Assistant》论文中的实验及其模型框架。论文提出了一种名为 LLaRA（Large Language-Recommendation Assistant）的创新方法，旨在结合传统推荐模型的行为模式和大语言模型（LLM）所具备的世界知识和推理能力，以提升顺序推荐的效果。顺序推荐任务指的是根据用户的历史互动序

列，预测用户下一次可能感兴趣的项目或内容。传统的推荐模型通常通过学习用户的历史行为数据（如商品点击、视频观看等）来进行预测，但这些模型仅依赖用户行为的序列信息，而忽视了对项目本身的更广泛知识。另一方面，大语言模型通过处理大量的文本数据，具备了较强的语言理解和推理能力，但它们在推荐系统中的应用还未得到充分探索。

复现此研究的目的是验证 LLaRA 在不同数据集上的表现，并深入分析其与传统推荐模型和其他基于 LLM 的方法相比的优劣，特别是在 HitRatio@1 这一评价指标上的优势。同时，通过本次复现，我们也希望能为顺序推荐领域中 LLM 与传统推荐模型的结合提供更多的实验依据和理论支持。

2 相关工作

2.1 大型语言模型

自从 20 世纪 50 年代图灵测试被提出以来，人类一直在探索通过机器掌握语言智能。语言本质上是一种由语法规则支配的复杂、精细的人类表达系统。开发能够理解和掌握语言的能力的人工智能（AI）算法是一项重大挑战。作为一种主要方法，语言建模在过去二十年里被广泛研究，用于语言理解和生成，从统计语言模型发展到神经语言模型。最近，通过在大规模语料库上预训 Transformer 模型，提出了预训练语言模型（PLM），显示出在解决各种自然语言处理（NLP）任务中的强大能力。研究人员发现，模型扩展可以提高模型容量后，他们进一步通过增加参数规模到更大的尺寸来研究扩展效应。有趣的是，当参数规模超过一定水平时，这些扩大的语言模型不仅实现了显著的性能提升，还表现出一些特殊能力诸如上下文学习等在小型语言模型中不存在。为了区分不同参数规模的语言模型，研究界为具有显著规模，包含数十亿或数百亿参数的 PLM 创造了“大型语言模型”（LLM）这一术语。最近，学术界和工业界都大大推进了 LLMs 的研究，一个显著的进展是 ChatGPT 的推出，这引起了社会的广泛关注。LLMs 的技术演变对整个 AI 社区产生了重要影响，将彻底改变我们开发和使用 AI 算法的方式。诸如 GPT-4 [1] 和 Llama 等大型语言模型（LLMs）不仅表现出显著的性能提升，还展现了包括常识推理与指令遵循在内的新兴能力。此外，领域特定的大语言模型如金融 [12]、医学 [9] 和法律 [2] 领域通过结合领域专业知识与 LLMs 固有的常识知识，进一步增强了其性能。这些进展激发了我们对 LLMs 在推荐领域潜力的进一步研究。

2.2 大型语言模型应用于推荐系统

序列推荐旨在基于用户的历史交互序列预测下一步符合其偏好的项目。先前研究采用了多种复杂的模型架构如循环神经网络 [3] [8]、卷积神经网络 [11] [13] 和注意力机制 [10]，以更好地表征用户偏好。随着 LLMs 的兴起，其在序列推荐中的潜力受到越来越多的关注。一方面，LLMs 存储的丰富世界知识可以为物品提供丰富的背景信息；另一方面，LLMs 的推理能力能够增强下一个项目的预测。

当将 LLMs 集成到推荐系统中时，研究方法主要分为两类。第一类是将 LLMs 作为增强器，通过生成 LLM 标记或嵌入 [4] [5] [14] 来增强传统推荐模型。在这种方法中，LLMs 通常作为特征提取器或文本生成器，利用其在整合多种信息来源方面的能力。然而，实际的推荐过程仍由传统模型完成，LLMs 的推理能力未被充分挖掘。第二类方法是直接将 LLMs 作为

推荐器。这包括对 LLMs 进行从头训练、微调、提示设计及上下文学习 [6] [7]，使其直接用于推荐任务。

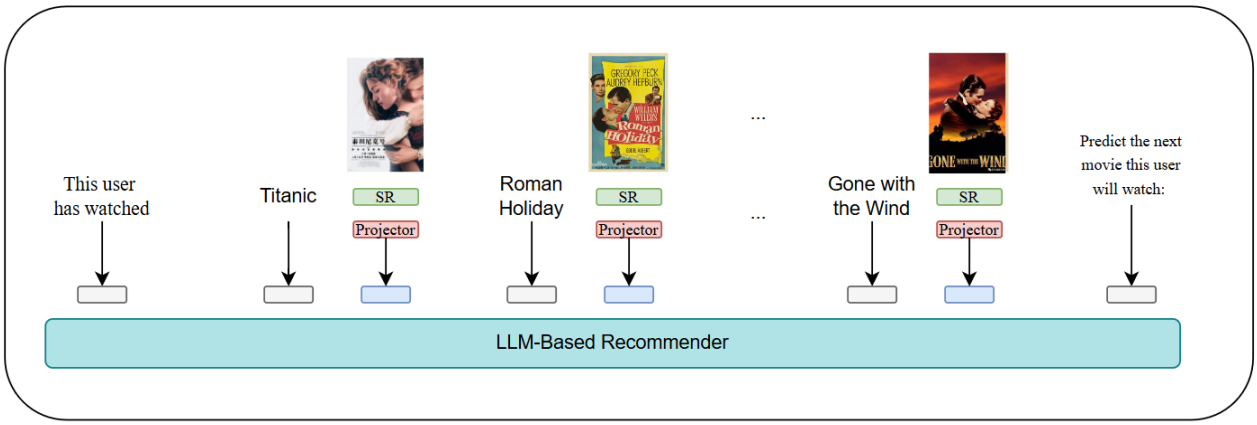
3 本文方法

3.1 本文方法概述

LLaRA 框架提出了一种新的混合提示方法（Hybrid Prompting），通过将传统推荐模型中的 ID 嵌入与项目的文本特征相结合，使得大语言模型能够更好地理解用户的行为模式和项目的语义特征。此外，LLaRA 还引入了课程学习（Curriculum Learning）策略，逐步将模型从简单的文本提示过渡到更复杂的混合提示，从而提高了训练效率和推荐质量。

3.2 混合提示方法（Hybrid Prompting）

将传统推荐模型的 ID 嵌入与项目的文本特征结合，形成多模态项目表示，充分利用用户行为和项目语义信息。投影器将 ID 嵌入映射到大语言模型（LLM）输入空间，确保行为标记与文本标记能够一起作为输入。



文本表示：物品的文本特征，如标题和描述，是利用 LLMs 中固有的常识知识的关键。正式来说，对于一个具有文本元数据 txt_i 的物品 i，可以通过以下方法获得文本表示：

$$< \text{emb}_{\text{t}}^{\text{i}} >= \text{LLM} - \text{TKZ}(\text{txt}_{\text{i}}) \tag{1}$$

其中 LLM-TKZ 代表 LLM 分词器和词嵌入层，封装了将文本元数据转换为标记表示的过程。

行为表示：在并行的情况下，传统的顺序推荐模型，如 GRU4Rec（Hidasi 等人，2016 年），Caser（Tang 和 Wang，2018 年），以及 SASRec（Kang 和 McAuley，2018 年），在历史交互数据上训练后，有效地捕捉了基于 ID 的项目嵌入中的顺序模式。正式地，对于项目 i，传统推荐模型学习到的基于 ID 的表示可以表示为：

$$\mathbf{e}_{\text{s}}^{\text{i}} = \mathbf{SR} - \mathbf{EMB}(\mathbf{i};\ _{\mathbf{e}}) \tag{2}$$

与可以自然插入提示并轻松被 LLMs 解释的项感知文本相比，基于 ID 的项表示可能与 LLM 提示的文本性质不兼容。因此，我们将基于 ID 的表示视为一种独立的模态，与文本数

据分开。为了弥合模态差距，将推荐器的基于 ID 的表示空间映射到 LLMs 的语言空间是至关重要的。这种对齐允许 LLMs 解释和利用传统推荐器提炼的行为知识。

为便于对齐，我们引入一个专用模块，使用两层感知机组成的可训练的投影器，作用是将基于 ID 的项目表示投影到 LLM 的空间中

$$< \mathrm{emb}_{\mathrm{s}}^{\mathrm{i}} >= \mathrm{Proj}(\mathrm{e}_{\mathrm{s}}^{\mathrm{i}};\ _{\mathrm{p}}) \tag{3}$$

混合表示：在获取文本表示和行为表示以及项目 i 后，对这两个表示进行整合得到最终的混合表示，混合表示能够结合两个表示进而全面描述项目 i。

$$< emb_c^i >= Concat(< emb_t^i >, < emb_s^i >) \tag{4}$$

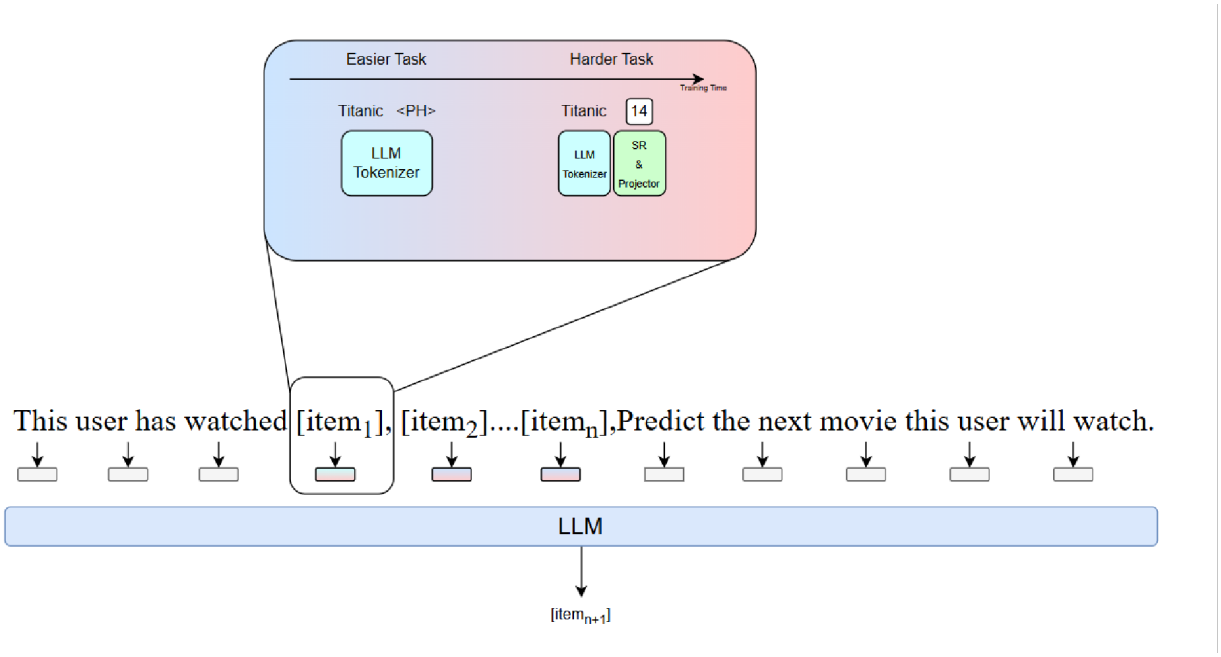
3.3 混合提示设计（Hybrid Prompt Design）

纯文本提示通过在提示中使用文本元数据来表示项目，每个文本源数据的 token 后都紧跟着一个占位符 [PH]；混合提示与纯文本提示相比则是将占位符的地方替换为从传统序列推荐模型中提取的 embedding。

(a) Text-only prompting method.	**(b)** Hybrid prompting method.
Input: This user has watched Titanic [PH], Roman Holiday [PH], ... Gone with the wind [PH] in the previous. Please predict the next movie this user will watch. The movie title candidates are The Wizard of Oz [PH], Braveheart [PH],..., Waterloo Bridge [PH],... Batman & Robin [PH]. Choose only one movie from the candidates. The answer is	**Input:** This user has watched Titanic $[\mathrm{emb}_s^{14}]$, Roman Holiday $[\mathrm{emb}_s^{20}]$, ... Gone with the wind $[\mathrm{emb}_s^{37}]$ in the previous. Please predict the next movie this user will watch. The movie title candidates are The Wizard of Oz $[\mathrm{emb}_s^{5}]$, Braveheart $[\mathrm{emb}_s^{42}]$,..., Waterloo Bridge $[\mathrm{emb}_s^{20}]$,... Batman & Robin $[\mathrm{emb}_s^{19}]$. Choose only one movie from the candidates. The answer is
Output: Waterloo Bridge.	**Output:** Waterloo Bridge.

3.4 课程学习策略（Curriculum Learning）

通过逐步引入不同复杂度的任务，在训练初期仅使用文本提示，逐渐过渡到混合提示，帮助模型从简单到复杂地学习推荐任务，提升训练效率并减少过拟合风险。



共设计了简单任务与复杂任务两个任务，简单任务对应为 3.3 节中的纯文本提示，复杂任务对应为 3.3 中的混合提示。根据这两个任务设计了课程学习策略的调度器，

$$p(\tau) = \frac{\tau}{T} \quad (0 \leq \tau \leq T) \tag{5}$$

累计总训练时间为 T。

4 复现细节

4.1 与已有开源代码对比

参考了 pmixer 发布的 SASRec.Pytorch 代码中产生嵌入的部分用于基于 SASRec 的 LLaRA 架构中来自传统序列推荐的行为嵌入部分;参考了 Balázs Hidasi 发布的 GRU4RecPyTorchOfficial 代码中产生嵌入的部分用于基于 GRU4Rec 的 LLaRA 架构中来自传统序列推荐的行为嵌入部分；参考了 graytowne 发布的 caser_pytorch 代码中产生嵌入的部分用于基于 Caser 的 LLaRA 架构中来自传统序列推荐的行为嵌入部分；参考了 ljy0ustc 发布的 LLaRA 代码中的模型架构部分。

4.2 实验环境搭建

实验基于 python，2.0.0 版本搭建，基于 Pytorch-Lighting 库编写，使用实验使用两张 NVIDIA A100 进行模型训练。

5 实验结果分析

复现实验采用了原文中的三个标准数据集：Steam、LastFM 以及 MovieLens。使用了 SASRec、GRU4Rec 以及 Caser 三个序列推荐模型作为为 LLaRA 提供行为嵌入的模型。实验中采用 Llama2-7b 作为 LLaRA 的骨干模型。实验的测试指标为 HitRatio@1 以及 Valid Ratio。HitRatio@1：表示模型是否能够正确预测用户下一步行为的准确率。对于每个用户，从候选集中的 10 个物品中选出预测结果，计算预测正确的比例。HitRatio@1 越高，模型的推荐准确度越高。Valid Ratio：由于 LLaRA 采用生成模型进行推荐，因此需要计算模型生成有效推荐的比例。有效推荐指的是模型生成的推荐项位于候选集内。如果模型生成的推荐项不在候选集中，则视为无效推荐。Valid Ratio 越高，表示模型能够更有效地按照输入提示生成合理的推荐项。

5.1 复现结果

LLaRA 在三个数据集上的表现相比于其它所有基线都有着比较显著的优势，不仅提升了推荐准确度，还显著提高了生成推荐的有效性，证明 LLaRA 的混合方法能够综合两者的优点，展现出更强的推荐能力。对于传统的顺序推荐器，它们的 HitRatio@1 得分低于 LLaRA，这些模型仅基于用户的行为模式进行预测，没有整合有关物体的相关信息，这也突出了在推荐过程中将关于物品的世界知识纳入的重要性。对于基于 LLM 的方法，MoRec 忽略了 LLMs

的推理能力，而 TALLRec 没有纳入传统的序列推荐器。这突出了需要一个更全面的方法，结合 LLMs 和传统推荐模型的优势的必要性。

		MovieLens		Steam		LastFM	
		ValidRatio	HitRatio@1	ValidRatio	HitRatio@1	ValidRatio	HitRatio@1
Traditional	GRU4Rec	1.0000	0.3510	1.0000	0.3748	1.0000	0.2260
	Caser	1.0000	0.3438	1.0000	0.3806	1.0000	0.2186
	SASRec	1.0000	0.3556	1.0000	0.3980	1.0000	0.2367
LLM-based	Llama2	0.4263	0.0568	0.1883	0.0364	0.2850	0.0231
	MoRec	1.0000	0.2634	1.0000	0.3856	1.0000	0.1674
	TALLRec	0.9203	0.3825	0.9588	0.4437	0.9638	0.4127
Ours	LLaRA(GRU4Rec)	0.9246	0.4338	0.9466	0.4638	0.9423	0.4123
	LLaRA(Caser)	0.9223	0.4515	0.9531	0.4619	0.9476	0.4156
	LLaRA(SASRec)	0.9412	0.4249	0.9683	0.4726	0.9536	0.4288

5.2 复现结果在 ValidRatio 指标上的对比

在 ValidRatio 这个指标上，复现结果与原文相比在数值上有着 3%-5% 的差距，在 Steam 数据集上使用 GRU4Rec 的 LLaRA 上差距最大，达到 0.0509；在 MovieLens 数据集上使用 SASRec 的 LLaRA 差距最小，为 0.0272。

	MovieLens		Steam		LastFM	
	ValidRatio	±	ValidRatio	±	ValidRatio	±
LLaRA(GRU4Rec)	0.9246	-0.0438	0.9466	-0.0509	0.9423	-0.0413
LLaRA(Caser)	0.9223	-0.0461	0.9531	-0.0435	0.9476	-0.0442
LLaRA(SASRec)	0.9412	-0.0272	0.9683	-0.0292	0.9536	-0.0464

5.3 复现结果在 HitRatio@1 指标上的对比

在 HitRatio@1 这个指标上，复现结果与原文相比在数值上有着 3.5%-5% 的差距，在 Steam 数据集上使用 GRU4Rec 的 LLaRA 上差距最大，达到 0.0286；在 MovieLens 数据集上使用 SASRec 的 LLaRA 差距最小，为 0.0172。

	MovieLens		Steam		LastFM	
	HitRatio@1	±	HitRatio@1	±	HitRatio@1	±
LLaRA(GRU4Rec)	0.4338	-0.0183	0.4638	-0.0286	0.4123	-0.0221
LLaRA(Caser)	0.4515	-0.0222	0.4619	-0.0255	0.4156	-0.0188
LLaRA(SASRec)	0.4249	-0.0172	0.4726	-0.0223	0.4288	-0.0220

6 总结与展望

本报告详细复现了《LLaRA: Large Language-Recommendation Assistant》论文中的实验和模型框架，成功验证了 LLaRA 在顺序推荐任务中的有效性。通过引入混合提示方法（Hybrid Prompting）和课程学习策略（Curriculum Learning），LLaRA 能够将用户的行为模式与项目的语义信息有效结合，从而提升推荐的准确性和鲁棒性。在多个标准数据集（MovieLens、Steam 和 LastFM）上的实验结果表明，LLaRA 在 HitRatio@1 等指标上显著优于传统的顺序推荐模型（如 SASRec、GRU4Rec 和 Caser），并且其生成的推荐具有更高的有效性。

LLaRA 框架通过将传统推荐模型与大语言模型的优点结合，为顺序推荐任务提供了一种创新的解决方案。复现过程中，我们验证了 LLaRA 在推荐精度、生成有效推荐以及计算资源效率方面的优势，为未来在推荐系统领域中引入更复杂的信息处理和推理能力提供了有力支持。

尽管 LLaRA 框架在实验中表现优异，但仍存在一些改进空间和挑战，未来的研究可以从以下几个方面进行扩展：LLaRA 的训练过程是离线进行的，未来的研究可以考虑如何将其应用于实时推荐系统。通过增量学习或在线微调技术，使得模型能够在用户行为发生变化时实时更新推荐结果，从而提高系统的动态适应能力；尽管 LLaRA 框架通过 LLM 和传统推荐模型的结合提升了推荐性能，但其黑箱性质依然存在。未来的研究可以聚焦于提升 LLaRA 模型的可解释性，使得推荐系统能够更加透明地向用户解释推荐结果，增强用户的信任感。

参考文献

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[2] Jiaxi Cui, Zongjian Li, Yang Yan, Bohua Chen, and Li Yuan. Chatlaw: Open-source legal large language model with integrated external knowledge bases. *CoRR*, 2023.

[3] B Hidasi. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*, 2015.

[4] Yupeng Hou, Zhankui He, Julian J McAuley, and Wayne Xin Zhao. Learning vector-quantized item representation for transferable sequential recommenders. corr abs/2210.12316 (2022), 2022.

[5] Yupeng Hou, Shanlei Mu, Wayne Xin Zhao, Yaliang Li, Bolin Ding, and Ji-Rong Wen. Towards universal sequence representation learning for recommender systems. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 585–593, 2022.

[6] Jiacheng Li, Ming Wang, Jin Li, Jinmiao Fu, Xin Shen, Jingbo Shang, and Julian McAuley. Text is all you need: Learning language representations for sequential recommendation.

In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1258–1267, 2023.

[7] Junling Liu, Chao Liu, Peilin Zhou, Renjie Lv, Kang Zhou, and Yan Zhang. Is chatgpt a good recommender? a preliminary study. *arXiv preprint arXiv:2304.10149*, 2023.

[8] Massimo Quadrana, Alexandros Karatzoglou, Balázs Hidasi, and Paolo Cremonesi. Personalizing session-based recommendations with hierarchical recurrent neural networks. In *proceedings of the Eleventh ACM Conference on Recommender Systems*, pages 130–137, 2017.

[9] Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023.

[10] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 1441–1450, 2019.

[11] Jiaxi Tang and Ke Wang. Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the eleventh ACM international conference on web search and data mining*, pages 565–573, 2018.

[12] Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*, 2023.

[13] Fajie Yuan, Alexandros Karatzoglou, Ioannis Arapakis, Joemon M Jose, and Xiangnan He. A simple convolutional generative network for next item recommendation. In *Proceedings of the twelfth ACM international conference on web search and data mining*, pages 582–590, 2019.

[14] Zheng Yuan, Fajie Yuan, Yu Song, Youhua Li, Junchen Fu, Fei Yang, Yunzhu Pan, and Yongxin Ni. Where to go next for recommender systems? id-vs. modality-based recommender models revisited. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2639–2649, 2023.