Recommendation as Instruction Following: A Large Language Model Empowered Recommendation Approach

作为指令遵循的推荐：一个大语言模型赋能的推荐方法

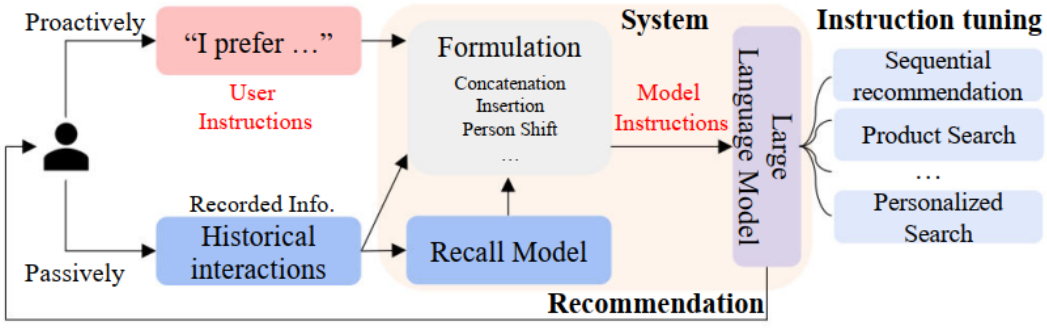
太长不看版：

研究团队开发基于 LLM 的推荐系统，其将推荐任务视为 LLM 的指令遵循，设计了通用的指令格式并将其进一步填充，随后将其用于 LLM 的指令调优，以使 LLM 适应推荐任务，实验结果表明该方法最终取得了不错的效果。

摘要

推荐系统在过去几十年中受到了研究和工业界的广泛关注，大量的研究致力于开发有效的推荐系统。基本上，这些模型主要从历史行为数据中学习用户的潜在偏好，然后估计用户-物品的匹配关系用于推荐。

受到最近大语言模型的发展的启发，研究团队采用的一种不同的开发推荐模型的方法，将推荐视作 LLM 的指令执行问题。其核心思想是，用户的偏好或需求可以用自然语言描述（称为指令），所以 LLM 可以理解并进一步执行指令，以完成推荐任务。研究团队对开源的 LLM（3B Flan-T5-XL）进行了指令调优，以使 LLM 更好地适应推荐系统。为此，研究团队首先设计了一种通用的指令格式，用自然语言描述用户的偏好、意图、任务形式和上下文。然后研究团队手工设计了 39 种指令模板，并自动生成了大量具有不同类型偏好和意图的用户个性化指令数据。为了展示方法的有效性，研究团队将指令模板实例化为几个广泛研究的推荐（或搜索）任务，并在这些任务上使用真实数据集进行了广泛的实验评估。实验结果表明，该方法可以在这些评估任务上超越几个有竞争力的基准方法。这个方法为开发更用户友好的推荐系统提供了一个新的视角和见解，用户可以通过自然语言指令与系统自由交流，并获得更准确的推荐结果。



1. 方法

1.1 指令格式

研究团队首先设计了一个合适的指令格式，以有效地描述用户的目的，偏好和任务形式。具体来说，一个指令由以下几个部分组成

偏好 Preference (P)：描述用户对物品属性特征的天生和长期形成的个性化品味

意图 Intention (I)：描述用户短期想要达到或完成的目标或行动。

任务形式 Task Form (T)：描述用户期望得到或提供的结果类型和数量。

Preference	Intention	Task Form
None (Cold-start users)	None (Exploratory interaction)	Pointwise (Discriminate a candidate)
Implicit (User's context info.)	Vague (Ambiguous idea of target)	Pairwise (Compare a item pair)
Explicit (Explicit expression)	Specific (Clear idea of target)	Matching & Reranking (Retrieving the candidates, refining the ranking)

有了这个说明格式，研究团队可以实例化不同的互动场景。

1.2 指令生成

对于每个用户，研究团队采用教师 LLM 基于其历史互动记录和评论生成个性化的自然语言提示。这些提示突出表现了用户的偏好、意图和上下文信息。然后，根据指令格式，研究团队将这些提示组合成完整的指令以执行各种推荐任务。

标注指令各部分

为了更好地通过自然语言指令格式化用户需求的表达，首先根据不同交互场景的实例化手工创建不同的粗粒度模板。然后，进一步用从交互数据中提取或由教师-LLM 生成的具体用户偏好和意图（指细粒度的用户个性化指令）来填充这些模板。以下，详细说明构建过程。

偏好标注：研发团队使用不同的策略来生成用户偏好，考虑到不同程度的个性化。对于隐式偏好 P_1，将项目的标题作为其表示，并利用用户的历史交互来填充指令模板，它们可以实例化为“用户之前购买过以下项目：*{[MASK]}*”。而对于显式偏好 P_2，由于数据集中通常没有用户偏好的显式描述，使用教师-LLM（即 GPT-3.5）来扮演用户，并根据历史交互生成偏好的显式表达。GPT-3.5 显示出了强大的文本理解能力，它可以根据历史交互行为生成关于用户偏好或意图的非常有意义的表达。下面是一个例子，描述了 GPT-3.5 生成用户偏好的过程：

[原始行为序列]：“1. 生化危机：启示录 2 - PlayStation 4 → 2. 生化危机 4 - PlayStation 4 标准版。”

[生成的显式偏好]：“他喜欢基于恐怖的游戏，有强烈的叙事性。”

意图标注：类似地，可以生成不同清晰度的用户意图，包括模糊意图 I_1 和具体意图 I_2。为了得到模糊意图，由于评论提供了关于用户个人品味和进行特定交互原因的宝贵证据，研究团队考虑从目标评论（与目标项目相关联的评论）中提取意图。特别地，研究团队使用教师-LLM 来处理这些评论，并从中提取意图。下面是一个例子，描述了从评论文本中提取用户意图的过程：

[原始目标评论]：“我的儿子喜欢…这个游戏。我很高兴我给他买了这个。”

[生成的模糊意图]：“我喜欢给我儿子买他喜欢的游戏。”

至于具体意图，用户在使用推荐系统时有时会对某些项目有明确的需求。目标项目的类别提供了反映用户真实意图的重要证据。因此，从目标项目的类别信息中提取用户的具体意图，通过将多个相关类别标签拼接成一个意图表达：

[生成的具体意图]：“视频游戏、PC、配件、游戏鼠标。”

与从用户历史交互中提取的模糊意图相比，提取出来的意图更具体，并且可以直接反映用户的真实意图。

任务形式标注：在本文中，研究团队主要关注三种任务形式:T_0、T_2 和 T_3。具体来说，对于点评价推荐 T_0，研究团队将说明表述为:“基于<与用户相关的信息>，用户是否有可能与<目标项目>下一步互动?”，系统应该回答“是”或“否”。对于匹配任务 T_2，说明概述为“预测下一个可能的项目”。而对于重新排序任务 T_3，一组候选者被纳入说明中:“从以下<候选者>中选择一项”。尽管目前研究团队没有包括成对推荐 T_1，但它可以很容易地集成到研究团队提出的框架中。

增加指令的多样性

最近的研究证明了增加说明数量和多样性的有效性。因此，为了进一步提高说明调整的性能，研究团队提出使用几种策略来增加说明数据的多样性。

任务反转（Turn the task around）这种策略是指普通说明的输入和输出的互换。

[普通说明]:“用户搜索<查询>，你能生成响应他查询的项目吗?”

[交换说明]:“用户想买:<目标项目>，但他不知道如何表述查询，请帮助他生成查询。”

加强偏好和意图之间的相关性(Enforcing the relatedness between preference and intention)这是指说明中透露的短期用户意图与长期偏好应高度相关。

[意图→偏好]:“用户搜索<查询>并选择产品<目标项目>。基于他的查询和选择，请推断他的历史互动。”

[偏好→意图]:“用户有以下<历史互动>，你可以推断出他们的偏好。请根据他的偏好预测用户的下一个查询和他可能购买的项目。”

连锁思维(CoT)样推理(Chain-of-thought like reasoning)它是指从用户的隐含行为到明确的偏好或意图的推理过程，最终导致推荐:

[CoT 说明]:“根据用户的<历史互动>，请推断他的偏好并推荐适合他的项目。”

[理想回应]:“根据用户的历史购买记录和搜索查询,我们可以看出用户喜欢<类别 1>和<类别 2>的商品。基于此,我们推荐<目标项目 1>和<目标项目 2>给用户。”

指令数据的分析和质量:

Quality Review Question	Preference	Intention
Is the instruction generated from the user's related information?	93%	90%
Does the teacher-LLM provide related world knowledge?	87%	22%
Does the instruction reflect the user's preference/ intention?	88%	69%
Is the instruction related to target item?	48%	69%

Statistic	
# of fine-grained instructions	252,730
- # of user-described preferences	151,638
- # of user intention in decision making	101,092
ave. instruction length (in words)	23.5
# of coarse-grained instructions	39
- # of preferences related instructions	17
- # of intentions related instructions	9
- # of combined instructions	13
ave. instruction length (in words)	41.4

1.3 指令调优

骨干 LLM

研发团队采用 3B Flan-T5-XL 作为骨干模型。由于 Flan-T5 已经基于 T5 用大量的指令数据进行了微调，它具有遵循自然语言指令的优异能力但其不具备优异的推荐功能。

训练和推理（*Training and Inference*）

1.推荐指令优化（Instruction Tuning for Recommendations）

有了生成的指令数据，可以通过指令调优来优化 LLM，这本质上是一种有监督的微调方式。具体来说，首先根据不同类型的指令标注期望的系统响应（目标输出）。例如，在指导模型预测下一个项目时，目标输出被标注为目标项目。而对于 CoT 类似的指令，目标输出被标注为用户对特定交互的推理过程。损失函数如下，其中 Y_k 为第 k 个实例对应的目标输出，I_k 为第 k 个实例对应的指令，B 是 batch size：

$$\mathcal{L} = \sum_{k=1}^{B} \sum_{j=1}^{|Y_k|} \log P\left(Y_{k,j} \mid Y_{k,<j}, I_k\right)$$

2.推理（Inference）

考虑到计算效率和模型容量，研究团队将 LLM 作为一个重排器，根据用户的指令对候选项目进行最终排序。在实际系统中，用户的需求非常多样化，研究团队期望指令可以通过利用自然语言的通用性有效地捕捉不同的偏好或意图。具体来说，在向用户提供推荐服务时，系统将首先根据用户指令（即用户发出的指令）和其他有用信息（例如历史交互）选择合适的粗粒度指令模板。然后，通过使用拼接、插入、人格转换等操作，将原始表达转换为模型指令。最后，推荐器（即 LLM）被要求执行指定用户需求的模型指令。然而，由于 LLM 中生成过程固有的随机性，存在生成候选集之外的项目的潜在风险。为了避免这个问题，研究团队直接将候选项目作为输入送入模型的解码器，并计算它们的可能性来确定推荐的最终排序。

2. 实验

2.1 实验设置

数据集

研究团队使用亚马逊数据集的“视频游戏（video games）”数据子集评估了模型在理解以用户为中心的指令上的表现，使用 CDs&Vinyl”子集评估了其对前所未

见的新数据的泛化能力。研究团队对所有数据集筛选不受欢迎的用户和交互次数少于五次的物品。由于 Flan-T5（基础模型）的上下文限制为 512tokens，研究团队将生成的行为序列截断为最多 20 个物品。

Dataset	#Users	#Items	#Inters	#Sparsity	Avg.*len*
Games	50,546	16,859	410,907	99.952%	8.13
CDs	93,652	63,929	871,883	99.985%	9.31

评估

研究团队采用 top-K 命中率（HR）和 top-K 归一化折现累积增益（NDCG）来评估性能。在本文中，K 被设置为 1、3 和 5。对于诸如顺序推荐和个性化搜索等交互场景，研究团队采用留一法策略进行评估。对于产品搜索等交互场景，研究团队将所有物品及其相关查询分为训练集（80%）、验证集（10%）和测试集（10%）。在这种设置下，验证集和测试集的实例在训练阶段是未见的，这增加了推断的难度。我们使用流行的开源推荐库 RecBole 实现了一些基线。值得注意的是，对于不同类型的粗粒度指令，研究团队评估在验证集中表现最佳的指令结果。将模型视为重新排名器，研究团队在测试集上对每个实例的真实值进行排名，其中包括九个随机抽样的负样本，用于评估，最后报告所有测试实例的平均得分。

2.2 不同用户信息需求的总体表现

下表介绍了不同场景下的这些实验，包括基线和性能比较，需要注意的是，由于这个模型的高成本，研究团队随机抽样了 500 个实例来评估其性能。尽管这可能不可避免地引入一些随机性，但研究团队认为在这个设置下得出的结论仍然可参考，以探索通用 LLM 的推荐能力。

Performance on sequential recommendation.

Methods	$\langle P_1, I_0, T_3 \rangle$				
	HR@1	HR@3	HR@5	NDCG@3	NDCG@5
BERT4Rec	0.5747	0.8188	0.9083	0.7176	0.7546
SASRec	0.6663	0.8741	0.9389	0.7887	0.8155
GPT-3.5	0.3640	0.6300	0.7300	0.5216	0.5623
InstructRec	**0.6947**	**0.8793**	**0.9429**	**0.8033**	**0.8295**
Improv.	+4.26%	+0.59%	+0.43%	+1.85%	+1.72%

Performance on product search.

Methods	$\langle P_0, I_1, T_3 \rangle$				
	HR@1	HR@3	HR@5	NDCG@3	NDCG@5
DSSM	0.7279	**0.9484**	**0.9899**	0.8587	0.8760
GPT-3.5	0.6700	0.9140	0.9480	0.8177	0.8399
InstructRec	**0.8263**	0.9411	0.9728	**0.8944**	**0.9075**
Improv.	+13.52%	–	–	+4.16%	+3.60%

Performance on personalized search. We have evaluated on three types of queries built from P_2, I_1, and I_2.

Methods	$\langle P_1, P_2, T_3 \rangle$					$\langle P_1, I_1, T_3 \rangle$					$\langle P_1, I_2, T_3 \rangle$				
	HR@1	HR@3	HR@5	NDCG@3	NDCG@5	HR@1	HR@3	HR@5	NDCG@3	NDCG@5	HR@1	HR@3	HR@5	NDCG@3	NDCG@5
TEM	0.5005	0.7368	0.8462	0.6383	0.6833	0.5723	0.7860	0.8712	0.6974	0.7325	0.8665	0.9149	0.9293	0.8954	0.9013
GPT-3.5	0.2740	0.5500	0.7000	0.4366	0.4979	0.3660	0.6360	0.7566	0.5256	0.5795	0.4580	0.7240	0.8100	0.6138	0.6499
InstructRec	**0.6959**	**0.8806**	**0.9452**	**0.8045**	**0.8312**	**0.8278**	**0.9444**	**0.9741**	**0.8971**	**0.9094**	**0.9269**	**0.9868**	**0.9951**	**0.9631**	**0.9665**
Improv.	+39.04%	+19.52%	+11.70%	+26.04%	+21.64%	+44.64%	+20.15%	+11.81%	+28.63%	+24.15%	+6.97%	+7.86%	+7.08%	+7.56%	+7.23%

2.3 若干深入问题

区分困难负面物品候选

为了评估模型在重新排名更实际的候选物品上的能力，这些物品是由强大的匹配模块检索得到的，与随机负面物品相比，更具挑战性，我们模拟了推荐系统中真实的匹配-重新排名流程。

为此，研究团队引入了一个匹配模块，从所有物品中检索候选物品列表，然后研究团队指示我们的模型在顺序推荐的情景中重新排名这些候选物品。在这个实验中，研究团队检索了与正目标物品一起的九个负面候选物品。实验结果表明其模型在从类似物品中区分和选择更符合用户信息需求的物品方面具有强大的能力。

Methods	Matching Module: $\langle P_0, I_1, T_3 \rangle$				
	HR@1	HR@3	HR@5	NDCG@3	NDCG@5
SASRec	0.0355	0.3448	0.5536	0.2127	0.2984
GPT-3.5	0.1100	0.3480	0.5020	0.2481	0.3113
InstructRec	**0.1841**	**0.4823**	**0.6648**	**0.3558**	**0.4307**
Improv.	+67.36%	+38.59%	+20.09%	+43.41%	+38.36%

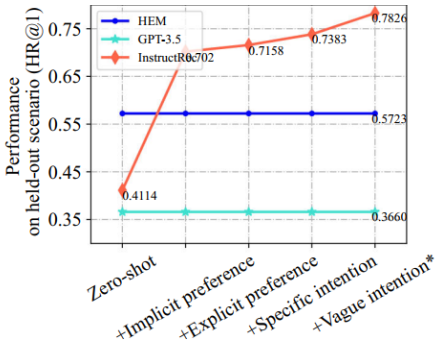
区分更多候选物品

在现实的推荐系统中，很常见遇到一个数以百计的物品池。为了评估其模型在更广泛的候选物品范围内的区分能力，研究团队模拟了个性化搜索的重新排名场景。这涉及对 99 个负面物品和目标物品进行随机抽样，得到了 100 个候选物品的集合。值得注意的是，由于受到上下文长度的限制，模型在同时处理这些候选物品时面临着一些限制。因此，研究者将一百个候选物品随机分成十组，并指示模型从每组中选择最有潜力的物品。然后，得到的十个物品将被重新排名，形成我们模型的最终排名结果。

Methods	$\langle P_1, I_1, T_3 \rangle$				
	HR@1	HR@3	HR@5	NDCG@3	NDCG@5
TEM	0.2794	0.4484	0.5284	0.3777	0.4106
InstructRec	**0.3276**	**0.7694**	**0.8334**	**0.5903**	**0.6163**
Improv.	+17.25%	+71.59%	+57.72%	+56.29%	+50.10%

指令的影响

研究团队将带有模糊意图的个性化搜索场景作为保留场景，并不断添加各种类型的细粒度指令进行指令调整。请注意，研究团队将“vague intention*”指的是从目标评论中模拟出的用户模糊意图的另一种表达方式。也就是说，由于评论包含了用户偏好和物品特征，研究团队使用教师 LLM 从物品和用户的角度分析用户的模糊意图，作为指令调整的训练数据和评估的测试数据。



跨数据集的泛化

下面评估模型对未见数据集的泛化能力，这些数据集可能与源数据具有不同的模式。为此，研究团队评估了模型从“Games”数据集泛化到“CDs”数据集的能力。

Methods	$\langle P_1, I_0, T_3 \rangle$				
	HR@1	HR@3	HR@5	NDCG@3	NDCG@5
BERT4Rec	0.5867	0.8226	0.9114	0.7249	0.7615
SASRec	**0.6713**	**0.8754**	**0.9428**	**0.7915**	**0.8194**
GPT-3.5	0.1380	0.3140	0.4780	0.2401	0.3067
Flan-T5-XL	0.0927	0.2886	0.4814	0.2035	0.2823
$\text{InstructRec}_{Games}$	0.2332	0.4392	0.6052	0.3511	0.4190

3 未来的工作

进一步扩大 LLMs 的规模进行指令调整，并考虑扩展上下文长度以建模长行为序列。此外，考虑将当前方法应用于多轮交互场景，在这种场景中，用户可以通过对话的方式与系统进行沟通。