arXiv:2603.07725v1 [cs.IR] 8 Mar 2026

面向基于大语言模型的生成式推荐的可验证推理

Xinyu Lin[1]，**Hanqing Zeng**[1]，**Hanchao Yu**[1]，**Yinglong Xia**[1]，**Jiang Zhang**[1]，**Aashu Singh**[1]，**Fei Liu**[1]，**Wenjie Wang**[2]，**Fuli Feng**[2]，**Tat-Seng Chua**[2]，**Qifan Wang**[1]

[1]Meta现代推荐系统（MRS） [2]新加坡国立大学

大语言模型（LLM）中的推理近期展现出巨大潜力，可通过深入理解复杂的用户偏好来提升生成式推荐。现有方法采用先推理后推荐范式，即由大语言模型在生成物品之前进行逐步推理。然而，由于缺乏中间验证，这一范式不可避免地会出现推理退化（即推理同质化或错误累积），从而损害推荐效果。为弥补这一不足，我们提出一种新颖的**推理-验证-推荐**范式，将推理与验证交织进行，以提供可靠的反馈，引导推理过程更忠实地理解用户偏好。为实现有效验证，我们确立了验证器设计的两项关键原则：1）可靠性 确保准确评估推理的正确性，并生成信息性指导；2）多维性 强调针对用户偏好的多个维度进行全面验证。据此，我们提出了一种有效的实现方案VRec。该方案采用验证器混合来确保多维性，同时利用代理预测目标来实现可靠性。在四个真实世界数据集上的实验表明，VRec在不牺牲效率的情况下，显著提升了推荐效果和可扩展性。

日期： 2026年3月10日
联系邮箱： Xinyu Lin：xylin1028@gmail.com，Qifan Wang：wqfcr@meta.com**代码：** https://github.com/Linxyhaha/Verifiable-Rec

Meta

1 引言

在大语言模型（LLM）中激励推理能力已得到广泛探索，并取得了显著成功，代表性成果包括 OpenAI-o1（OpenAI，2025年）、DeepSeek-R1（Guo 等，2025年）、Kimi-K2（Team 等，2025年）和 Qwen-QwQ（Team 等，2024年）。大语言模型在生成回答之前进行逻辑性的逐步推理（例如思维链（Wei 等，2022年）），显著增强了其处理规划（Wei 等，2025年）、编程（Ding 等，2024a）和数学（Huang 和 Yang，2025年）等复杂任务的能力。在推荐领域，大语言模型推理有望更有效地捕捉不断变化且难以简单概括的用户兴趣，以及多样化的物品特征（Lin 等，2024年）。鉴于此，利用推理能力实现基于大语言模型的生成式推荐（Reason4Rec）已引发越来越多的研究（Tsai 等，2024年；Kim 等），并逐渐成为推动推荐系统发展的一个极具前景的方向。

从技术上看，给定用户的历史交互，大语言模型旨在生成符合用户偏好的下一个物品（Bao等人，2025年）。不同于仅捕捉用户浅层偏好的即时生成方式（Fang等人，2025年；Tang等人，2025a），Reason4Rec采用**先推理后推荐**范式，在该范式中，大语言模型通常首先在潜在空间中以自回归方式对用户历史记录进行推理[1]，，随后生成下一个推荐项目。在这一范式下，这些自回归推理步骤致力于从隐式反馈（即，历史交互）中提炼复杂的用户偏好，引导模型实现更深入的理解，从而产生恰当的推荐。尽管前景可期，现有的先推理后推荐

[1]其他相关工作，尤其是专为在语言空间中进行推理的评分预测而设计的工作，将在第 5 节讨论。

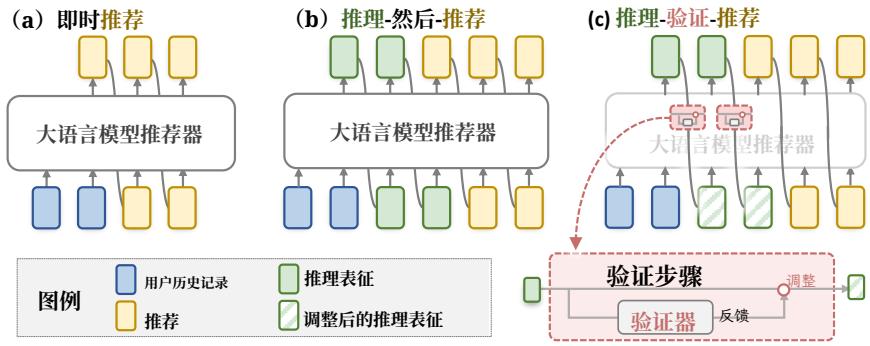


图1 （a）典型的基于大语言模型的推荐、（b）基于大语言模型的推荐推理，以及（c）我们提出的基于大语言模型的可验证推理范式对比。

生成范式面临一个根本性局限：未经验证的推理。在缺乏验证的情况下，大语言模型容易出现两类主要失败情形，从而导致推理退化：

- 同质化推理。由于中间推理步骤缺乏监督，推理过程可能停留在表层，反复依赖相同的虚假相关性，却无法发现新的见解（Yuan 等人，2024年；Ding 等人，2024b）（详细分析请参见第2节）。

- 误差累积推理。缺乏验证时，推理早期的失误可能会在后续步骤中传播并累积（Zhou 等人，2024年），最终放大错误并导致不可靠的推荐（见图3（b））。

鉴于此，我们提出一种新颖的可验证推理范式，使大语言模型能够依据验证反馈审查推理过程，从而更忠实地理解用户偏好，充分释放推理的潜力。具体而言，如图 1(c) 所示，可验证推理遵循 **推理-验证-推荐** 范式。该范式包含三个关键步骤：1）**推理步骤** 旨在审慎分析用户历史记录，从隐式反馈中挖掘潜在的用户偏好。基于中间推理结果，2）**验证步骤** 对推理结果进行论证并输出修正后的结果，供下一步推理使用。推理与验证步骤交错执行，使每次验证都能引导后续推理。最后，经过若干推理和验证步骤后，为了推荐物品，3）**推荐步骤** 基于推理结果自回归地生成下一个物品。

本质上，验证步骤连接各个推理步骤，评估推理并提供引导信号，以便进行调整，从而更准确地理解用户。由此自然引出一个关键研究问题：如何构建有效的验证器？我们提出验证器设计应遵循两项原则：

- **可靠性**确保推理输出得到恰当评估，并由验证器有效调整。然而，直接评估推理的正确性颇具挑战，因为推理步骤在潜在空间中进行，且没有天然存在的监督信号来表明其质量。因此，设计一项代理评估任务至关重要，使验证器能够提供反馈，以近似衡量每个推理步骤的质量。此外，验证器除了提供评估性反馈，还应生成引导信号，以有效修正中间推理表示，从而减轻同质化和错误累积推理造成的性能退化。

- **多维性**要求验证器从多个角度论证中间推理结果。仅依赖单一方面（例如，物品类别）不足以捕捉与丰富用户偏好的对齐。因此，我们考虑两种互补的维度多样性形式。用户内多样性用于论证不同方面（例如，语义信息和协同信息）上的多方面用户偏好对齐；与此同时，用户间多样性强调，不同用户可能关注物品的不同方面（例如，一位用户可能因为电影《泰坦尼克号》而喜欢歌曲《我心永恒》，另一位用户则可能因为演唱者而喜欢这首歌）。

基于上述原则，我们提出 VRec，这是可验证推理范式在基于大语言模型的推荐中的一种有效实现。为确保验证的多维性，我们设计了一种

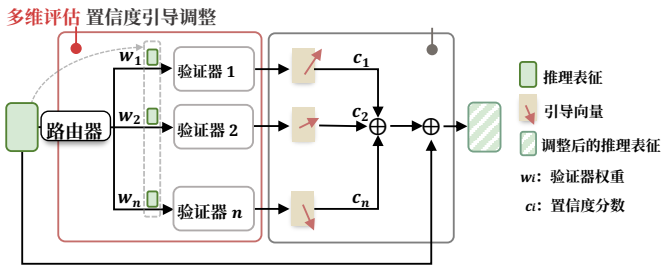


图2 “推理-验证-推荐”范式中验证步骤的示意图。多方面验证确保验证的多样性，而基于置信度的调整则强调验证的确定性。

如图所示的新型验证器混合2。具体而言，为实现用户内维度多样性，我们采用一组验证器，每个验证器专注于特定方面，例如物品类别、标题语义和协同信息。为实现用户间维度多样性，我们进一步引入个性化路由器，根据各用户的行为自适应地调整不同验证器的贡献。另一方面，为确保验证可靠性，我们将群体级偏好预测任务设计为验证器的代理目标，并利用验证器的内部信号，即 例如，预测熵和模型权重，分别作为推理调整的评估反馈和引导信号。为实现 VRec，我们采用两阶段训练策略训练验证器和大语言模型推荐器，并引入新颖的单调性正则化，以促进推理准确性逐步提升。在四个真实世界数据集上进行的大量实验验证了所提方法的有效性、效率和良好的可扩展性。

本文的主要贡献总结如下：

- 我们强调验证在大语言模型推理中的重要性，并为基于大语言模型的生成式推荐提出一种新颖的推理-验证-推荐范式，同时给出验证器设计的两项原则。
- 为实现这一范式，我们提出了一种有效的实现方案 VRec。它采用验证器混合来捕捉多样化的用户偏好，并对中间推理步骤提供可靠反馈，从而发挥大语言模型推理的潜力。
- 我们在四个真实世界数据集上进行了大量实验，以验证可验证推理范式的实用性，并证明 VRec 在有效性、可扩展性和多样性方面的表现。

2 任务定义

本节首先介绍 Reason4Rec 现有的先推理后推荐 范式，并揭示该范式存在的问题。随后，我们将阐述可验证推理范式的定义。

- **Reason4Rec 回顾。** 给定按时间顺序排列的用户历史交互 $X = [i_1, i_2, \ldots, i_L]$，其中 $i \in \mathcal{I}$ 是物品标识符（例如，物品标题），且 $L = |X|$，基于大语言模型的推荐旨在生成符合用户偏好的下一个物品 $y = i_{L+1}$。

先推理后推荐范式。为利用推理进行推荐，先前研究主要在生成下一个物品之前进行推理。形式化地，给定输入 $\boldsymbol{x}$，由 θ 参数化的大语言模型 $\mathcal{M}$ 首先生成一个推理序列：

$$R = (\boldsymbol{r_1}, \boldsymbol{r_2}, \ldots, \boldsymbol{r}_m), \quad \text{where } \boldsymbol{r}_t = \mathcal{M}(\cdot \mid X, \boldsymbol{r}_{1:t-1}; \theta), \tag{1}$$

其中，每个中间推理表示 $\boldsymbol{r}_j \in \mathbb{R}^{d_\mathcal{M}}$ 都是一个潜在表示（Zhang 等人，2025年c），其维度为大语言模型的隐藏维度 $d_\mathcal{M}$。在用户历史记录 X 和推理 R的共同条件下，大语言模型随后生成下一个物品：

$$\hat{y}_t = \arg\max_{v \in \mathcal{V}} \mathcal{M}(v \mid X, R, \hat{y}_{<t}; \theta), \tag{2}$$

其中 $\mathcal{V}$ 是物品词汇表，$\hat{y}_t$ 是$\hat{y}$中的第 t 个词元，$\hat{y}_{<t}$ 表示 y_t之前的词元序列。这种范式引入了额外的逐步推理，从而增强了对用户行为的理解，并带来更好的推荐。

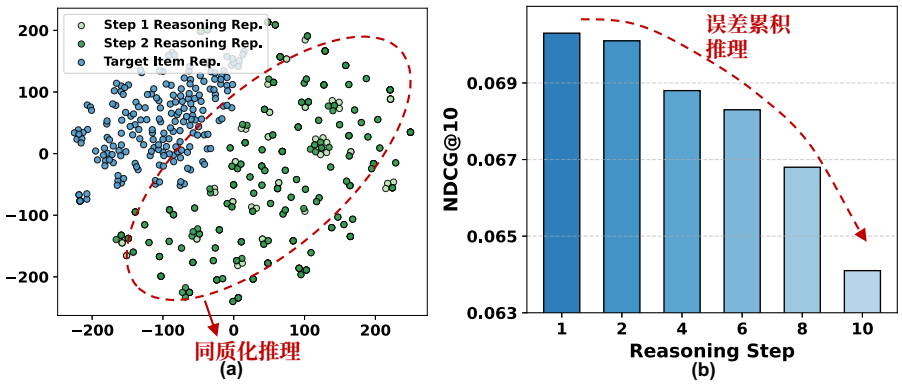


图 3 (a)展示了 LatentR^3 在包含 2 个推理步骤时，潜在推理表示和目标物品表示的 t-SNE 可视化结果。(b)展示了在先推理后推荐范式下，随着推理步骤增加，性能持续下降。

该范式存在的问题。尽管前景可观，但由于缺乏对中间推理结果的反馈，推理过程未经验证。这可能导致两个后果：

• 同质化推理。若缺少中间验证，大语言模型可能会走捷径跳过推理过程，并利用虚假相关性。不失一般性，大语言模型推荐器通过以下方式进行优化：

$$\arg\min_{\theta\in\Theta} \mathbb{E}_{X\sim\mathcal{D}}\mathbb{E}_{R\sim\mathcal{M}}\mathcal{L}_\theta(\hat{y}\mid X,R), \tag{3}$$

其中 $\mathcal{L}_\theta(\cdot)$ 可替换为监督微调中的训练目标形式（例如，生成损失（Zhu 等人，2024年））或强化学习中的（例如，策略梯度目标（You 等人，2025a））。然而，监督信号仅提供给推荐项目 $\hat{}y$，忽略了对R的直接优化。因此，模型可能会收敛到退化分布，坍缩为琐碎或同质化的推理模式（详见附录 7.1 中的详细解释）。为从实证上验证这一问题，我们将中间推理的嵌入可视化，如图 3（a）所示。

• 误差累积的推理。在缺少验证的情况下，自回归生成的推理表示会受到误差累积问题的影响。换言之，初始的不准确推理最终会导致不恰当的推荐。因此，大语言模型推荐器难以实现深度推理，阻碍了推理能力的充分发挥（图 3（b）中的实证证据）。

• **基于大语言模型推荐的可验证推理。**为克服这些局限，我们提出了一种新颖的推理-验证-推荐范式，在推理表示中加入了额外的验证步骤。形式上，对于步骤 t 中的任意中间推理表示 $\boldsymbol{r}\sim\mathcal{M}(X,R_{<t})$，有

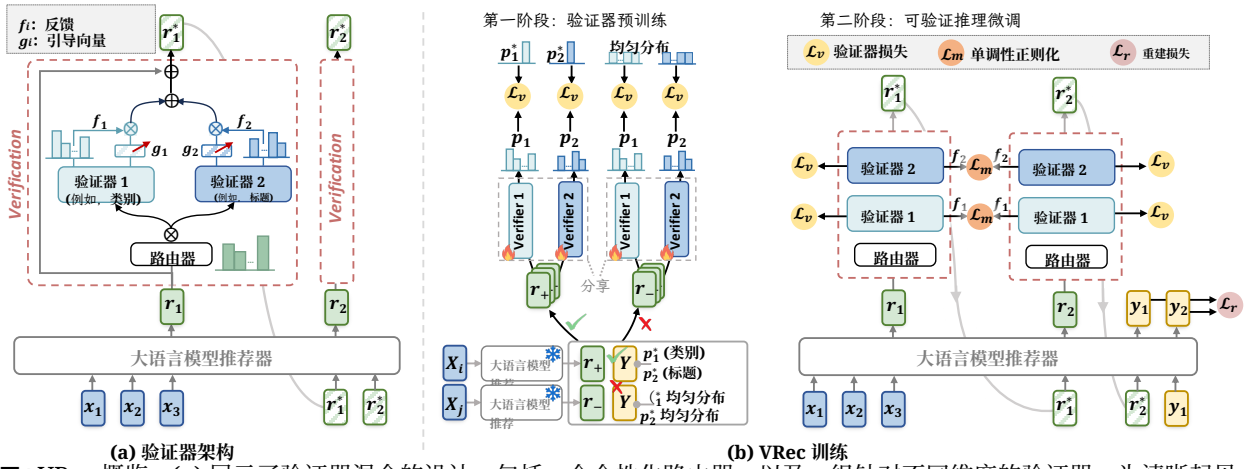
$$\begin{cases} f,g \leftarrow \text{Verifier}(\boldsymbol{r}), \\ \boldsymbol{r}^* \leftarrow \text{Adjuster}(\boldsymbol{r},f,g), \end{cases} \tag{4}$$

其中 f 和 g 分别表示评估反馈和引导信号，而 $\boldsymbol{r}^*$ 表示调整后的推理表示，随后它将取代初始的 $\boldsymbol{r}$，用于下一推理步骤，即 $R_{\le t}=(\boldsymbol{r}_1^*,\ldots,\boldsymbol{r}_t^*)$。经过 m 步推理与验证的交替过程后，即通过式（2）生成物品。

有效的可验证推理的关键在于验证器的设计。对此，我们提出两项基本原则：1）多维性旨在从多个方面评估 $\boldsymbol{r}$，确保用户内部及用户之间的偏好多样性。2）可靠性强调妥善设计评估反馈 f 和引导g，以有效反映推理质量和引导方向，从而实现可靠的推理调整。

3 VRec

为实现这两个目标，我们提出了一种有效的方案 VRec。如图4所示，VRec 采用一组验证器模型来评估推理结果，并结合反馈 f 和



(a) 验证器架构　　(b) VRec 训练

图4 VRec 概览。(a) 展示了验证器混合的设计，包括一个个性化路由器，以及一组针对不同维度的验证器。为清晰起见，我们以两个验证器为例进行说明。(b) 展示了 VRec 的两阶段训练策略，包括验证器预训练和可验证推理微调。

引导信号（第 3.1节）。为训练验证器，我们提出了一种两阶段训练策略，包括验证器预训练和可验证推理微调（第 3.2节）。

3.1 验证器架构

为了发挥可验证推理范式的潜力，我们旨在设计一个着重考虑两个关键因素的验证器，即多维性和可靠性。其架构如图4(a) 所示。

• **验证器混合**。为实现用户内维度多样性，即用户偏好可以具有多个维度，我们采用一组验证器 $\mathcal{V} = \{V_1, V_2, \ldots, V_n\}$，其中每个验证器 V_i 专注于某个特定方面，例如类别和协同信息。对于每个验证器 V_i，它会评估中间推理表示与用户在特定方面的偏好 $i \in \{1, \ldots, n\}$之间的一致性：

$$f_i, g_i \leftarrow V_i(\boldsymbol{r}), \tag{5}$$

其中，f_i 和 g_i 是针对方面 i的评估反馈和引导。反馈和引导集合分别记为 $\mathcal{F} = \{f_1, \ldots, f_n\}$ 和 $\mathcal{G} = \{g_1, \ldots, g_n\}$，随后将用于推理调整。

个性化路由器。除了用户内维度多样性，我们还考虑用户间维度多样性，即，不同用户可能会优先关注不同方面。例如，有些用户可能更关注物品类别（例如，爵士音乐），而另一些用户可能更在意价格。因此，我们为验证器集合引入个性化路由器，以获取各个方面的个性化验证权重：

$$\begin{cases} \boldsymbol{w} = g(\boldsymbol{r_i}), \quad \text{where } \boldsymbol{w} = [w_1, \ldots, w_n] \in \mathbb{R}^n \\ f_i, g_i \leftarrow V_i(w_i \boldsymbol{r}), \end{cases} \tag{6}$$

其中，$g(\cdot)$ 是个性化路由器，可根据各用户的行为为不同验证方面分配自适应权重。

• **评估反馈的设计** f_i。为了验证中间推理结果 $\boldsymbol{r}$，一种直观的解决方案是将验证任务视为针对推理结果正确性的二元分类，即，f_i 被表示为0或1。然而，训练这样的验证器并不可行，因为很难从推荐数据中的隐式用户反馈（例如，点击）中获取推理正确性的真实标签。更糟糕的是，能够得出准确推荐的潜在推理过程在粒度和维度上可能各不相同，从而引入大量噪声，妨碍构建有效的二元分类器。

偏好预测任务。或者说，能够得出准确推荐的推理应与群体层级用户偏好相一致（即，用户的粗粒度兴趣，例如嘻哈音乐）。从这个意义上说，

我们不在任意粒度层级评估推理，而是着重识别并改进可能与用户广泛偏好相冲突的推理过程。因此，我们将一项偏好预测任务定义为验证器进行二元评估的替代方案：

$$\begin{cases} \boldsymbol{p} = V_i(w_i \boldsymbol{r}), \quad \text{where } \boldsymbol{p} = [p_1, \ldots, p_{d_i}], \\ \hat{P} = \arg\max_j p_j, \end{cases} \tag{7}$$

其中 $\boldsymbol{p} \in \mathbb{R}^{d_i}$ 是偏好预测的概率分布， d_i 是预定义的群体级偏好类别数， $\hat{}P$ 是预测的群体级偏好（例如，类别为“嘻哈音乐”。相应地，验证器 V_i 是一个可学习分类器，可通过线性头或多层感知机实现（分类器设计的详细讨论见第 4.3.3节）。

基于熵的评估。直观而言，对于经过良好训练的验证器[2]，，如果推理表示 $\boldsymbol{r}$ 与群体层级用户偏好 P^* 对齐，则 $P = P^*$；否则，两者不对齐。然而，实时推荐中未知 P^*，因此我们使用熵来估计偏好对齐情况：

$$f_i = H(\boldsymbol{p}) = -\sum_{j=1}^{d_i} p_j \log p_j. \tag{8}$$

熵通常表示模型确定性，广泛用于衡量置信度水平（Fu 等人，2025年；Farquhar 等人，2024年；Liu 等人，2024年）。因此，较小的 f_i 意味着推理与高置信度的偏好 P^* 相对齐；而较大的 f_i 则意味着推理的置信度较低，且可能与偏好 P^*相冲突。

• **引导信号的设计** g_i。为了调整中间推理结果，我们随后需要获取引导信号 $\boldsymbol{g}_i$，以相应地调整推理，从而防止同质化及误差累积的推理问题。具体而言，先前研究表明，输出预测由充当原型的嵌入向量所锚定（Morin and Bengio，2005；Li等人，2018；Zhou等人，2022）。受此启发，我们将验证器最后一层的权重用作偏好原型，以进行推理调整。我们得到

$$\boldsymbol{g}_i = \boldsymbol{W}^i_{:,j^*}, \quad j^* = \arg\max_j p_j, \tag{9}$$

其中$\boldsymbol{W}^i \in \mathbb{R}^{d_{\mathcal{M}} \times d_i}$ 是验证器最后一层的权重， $d_{\mathcal{M}}$ 是大语言模型的隐藏维度。

基于置信度的调整。引导向量 $\boldsymbol{g}_i \in \mathbb{R}^{d_{\mathcal{M}}}$ 随后通过以下方式调整推理表示：

$$\boldsymbol{r}^* = \sum_{i=1}^{n} (1 - c_i)\boldsymbol{r} + c_i \boldsymbol{g}_i, \quad \text{where } c_i = \frac{1}{f_i} \text{ is the confidence score.} \tag{10}$$

直观而言，对于与群体层级用户偏好对齐良好的推理，即较大的 f_i，我们力求强化当前路径上的推理，并积极提升该推理，从而缓解同质化问题；相反，若推理偏离用户偏好 P^*，我们旨在防止误差传播，并鼓励模型以初始推理结果与引导向量的适度组合来探索其他推理路径。

3.2 VRec 训练

VRec 架构包含两个可学习组件：即验证器和大语言模型推荐器。为确保验证器提供可靠的反馈，并与推荐器无缝集成，我们采用了图4（b）所示的两阶段训练策略。

• **第一阶段：验证器预训练**。验证器预训练的目标是训练一个验证器，使其能够正确预测可靠的推理表示应捕获的群体级偏好。通过教会验证器准确预测群体级偏好，我们可以获得可靠的评估反馈 f 和信息性指导 $\boldsymbol{g}$，从而进行有效调整。

预训练数据收集。 为了训练验证器，我们首先收集预训练数据。具体而言，我们利用预训练的大语言模型推理推荐器，为以下内容生成推理过程及相应的推荐：

[2]有关验证器训练的详细信息，请参阅第3.2 节。

所有样本 $(X,\ y) \in \mathcal{D}$。对于每个样本，如果成功生成目标下一项，我们就将其视为正样本，并收集推理结果 R，再将其与目标物品各个方面对应的偏好配对 P（例如，类别方面对应的"爵士乐"）[3]。相反，对于未能生成目标下一项的样本，即负样本，我们将推理结果 R 与 $\emptyset$配对。所得数据集记为 $\mathcal{D}_v = \{(R,\ \mathcal{P}) \mid \mathcal{P} = \{P_1, \ldots, P_n\}$ 或 $\emptyset\}$。

训练目标。 随后，通过最小化正样本上的交叉熵，同时最大化负样本上的熵来优化验证器。我们将损失函数定义为:

$$\mathcal{L}_v = \begin{cases} -\log \Pr(\hat{P} = P), & \text{if } P \neq \emptyset, \\ -\alpha \sum_{j=1}^{d_i} p_j \log p_j, & \text{if } P = \emptyset, \end{cases} \tag{11}$$

其中 $\hat{P}$ 是根据式(7)预测的偏好，且 α 是一个表示负样本权重的超参数。直观而言，我们鼓励验证器对与偏好一致的推理作出准确预测。相反，对于低质量推理，我们鼓励验证器输出均匀分布，即导致较大 f的高熵预测。

• **阶段二：可验证推理微调**。随后，基于预训练验证器和大语言模型推荐器，我们进一步以端到端方式联合微调这两个模型，使可验证推理最终能够导向恰当的推荐。具体而言，我们有

$$\mathcal{L}_r = -\frac{1}{|\mathcal{D}|} \sum_{\mathcal{D}} \sum_{t=1}^{|y|} \mathcal{M}_\theta(X, R, y_{<t}), \tag{12}$$

其中 $R = (\boldsymbol{r}_1^*, \ldots, \boldsymbol{r}_2^*)$，且每个 r^* 均根据式(10)得到。

单调性正则化。值得注意的是，在多个验证步骤中共享验证器也可能导致捷径学习（Geirhos 等人，2020），从而阻碍更深层次的推理能力。为解决这一问题，我们提出单调性正则化，以进一步确保推理结果逐步变得更加准确，其定义如下:

$$\mathcal{L}_m = \max(0, f_i^t - f_i^{t-1}), \tag{13}$$

其中 $f_i^t = H(\boldsymbol{p})$ 是第 t 步推理的评估反馈，其定义见公式(8)。通过惩罚连续步骤间熵的增加，该正则化促使推理过程逐渐与群体层级用户偏好保持一致。

总体损失。为确保验证器具备评估推理与用户偏好之间一致性的能力，我们将验证器损失（公式(11)）纳入正则化项。总而言之，第二阶段的总体损失为:

$$\mathcal{L} = \mathcal{L}_r + \beta \mathcal{L}_v + \gamma \mathcal{L}_m, \tag{14}$$

其中 β 和 γ 分别是用于平衡验证器损失和单调性正则化的超参数。

3.3 VRec 实例化

为了在预训练的推理型大语言模型推荐器上实例化 VRec，并设推理步骤为 m，我们首先通过公式(11)对验证器进行预训练，使其评估推理过程。随后，通过公式(14)联合训练验证器和大语言模型推荐器，以实现面向推荐的可验证推理。在推理阶段，VRec 通过公式(10)为每个中间推理结果提供依据并进行调整。在执行交错推理与验证步骤后，大语言模型推荐器通过公式(2)生成下一个物品。VRec 的详细实例化过程见附录中的算法 1 。

• **计算开销分析**。在我们的可验证推理范式中，整体复杂度由两部分组成。以每次前向传播中的乘加运算（浮点运算次数）计算，基于大语言模型的推荐器占据主导地位，其复杂度为 $\mathcal{O}(mLd^2)$，其中 m 是推理步骤数，L 是 Transformer 层数，d 是隐藏维度。验证模块增加了 $\mathcal{O}(mnkd^2)$ 的复杂度，因为每个步骤都会将推理嵌入通过一个个性化路由器和 n 个验证器，每个验证器均为 k 层多层感知机（MLP）。

[3]请参阅附录 7.1 以了解群体级偏好标注的详细实现。

表 1 VRec 与基线方法在四个真实世界数据集上的总体性能。最佳结果以粗体显示，次佳结果带下划线。 $*$ 表示在单样本 t 检验下，VRec 相较于最佳基线方法结果的提升具有统计显著性（p- 值 < 0.01 ）。

	CDs				MicroLens				Goodreads				乐器			
方法	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10	R@5	R@10	N@5	N@10
GRU4Rec	0.0481	0.0671	0.0365	0.0426	0.0036	0.0068	0.0021	0.0031	0.0190	0.0332	0.0117	0.0163	0.0766	0.0938	0.0553	0.0609
Caser	0.0549	0.0749	0.0403	0.0467	0.0036	0.0063	0.0020	0.0029	0.0214	0.0378	0.0131	0.0183	0.0775	0.0972	0.0544	0.0608
SASRec	0.0784	0.0984	0.0576	0.0641	0.0044	0.0069	0.0029	0.0037	0.0182	0.0316	0.0114	0.0157	0.0825	0.0972	0.0726	0.0772
TIGER	0.0381	0.0555	0.0269	0.0325	0.0052	0.0075	0.0023	0.0030	0.0202	0.0384	0.0116	0.0175	0.0760	0.0945	0.0557	0.0616
LETTER	0.0417	0.0619	0.0285	0.0350	0.0053	0.0104	0.0028	0.0045	0.0215	0.0377	0.0134	0.0185	0.0707	0.1012	0.0483	0.0583
SETRec	0.0627	0.0942	0.0454	0.0553	0.0094	0.0129	0.0055	0.0066	0.0106	0.0174	0.0066	0.0088	0.0786	0.0864	0.0721	0.0746
D3	0.0747	0.0940	0.0596	0.0659	0.0076	0.0134	0.0057	0.0076	0.0328	0.0534	0.0218	0.0285	0.0996	0.1188	0.0853	0.0915
LatentTTS	0.0730	0.0912	0.0578	0.0643	0.0069	0.0123	0.0050	0.0068	0.0312	0.0513	0.0206	0.0265	0.0911	0.1068	0.0801	0.0866
LatentR^3	0.0801	0.0978	0.0645	0.0703	0.0089	0.0160	0.0065	0.0088	0.0349	0.0565	0.0231	0.0301	0.1022	0.1243	0.0881	0.0951
VRec-1Step	0.0836	0.1070	0.0655	0.0730	0.0100	0.0154	0.0076	0.0093	0.0365	**0.0593**	0.0236	0.0310	0.1067	0.1269	0.0911	0.0976
VRec	**0.0963***	**0.1213***	**0.0782***	**0.0862***	**0.0121***	**0.0235***	**0.0085***	**0.0121***	**0.0366***	0.0582*	**0.0242***	**0.0312***	**0.1098***	**0.1301***	**0.0944***	**0.0997***

尽管增加了这部分开销，验证器仍然十分轻量，因为其计算主要集中于少量低维偏好类别（约 20 个），而且与大语言模型骨干网络相比，其规模可以忽略不计。时间成本分析也在第 4.3.4 节中进行了实证研究。

4 实验

本节开展实验，以回答以下研究问题：**RQ1**：VRec与基线方法相比表现如何？**RQ2**：VRec的不同组件（即多维验证器、个性化路由器、单调性正则化和验证器预训练）对性能有何影响？**RQ3**：增加推理步骤和验证器的数量时，VRec的表现如何？**RQ4**：验证步骤的计算开销如何，超参数敏感性又如何？

4.1 实验设置

4.1.1 数据集

我们在来自不同领域的四个真实世界数据集上开展实验，涵盖音乐、微视频和图书。具体而言，我们采用两个广泛使用的亚马逊评论数据集[4]，1）**CD** 和 2）**乐器**，它们涵盖用户与音乐专辑和乐器产品的交互，并包含标题和类别等丰富的文本元信息。此外，我们还使用微视频数据集 3）**MicroLens**[5]，该数据集包含大量用户交互，以及微视频的文本描述（Ni等人，2024年）。此外，我们采用 4）**Goodreads**[6]，这是一个图书推荐数据集，提供用户与图书的交互数据；其中每个物品均附有书名、描述和类别。对于每个数据集，我们依据时间戳按时间顺序排列用户交互，并按照 8:1:1 的比例将其划分为训练集、验证集和测试集。对于每个训练样本，我们遵循（Bao等人，2024年）的做法，将物品序列的最大长度限制为 10。四个数据集的统计信息汇总于附录的表4 中。

- **评估。** 为评估这些模型，我们采用广泛使用的 Recall@K（R@K) 和 NDCG@K(N@K) (Wang等人，2021)，使用 $K = 5$ 和 10 来评估推荐准确率。

4.1.2 基线方法

我们将 VRec 与多种有竞争力的基线方法进行比较，包括传统判别模型、基于大语言模型的生成模型以及大语言模型推理增强模型。具体而言，1）**GRU4Rec**（Hidasi等人，2016）使用门控循环单元（GRU）捕捉物品的序列模式。2）**Caser**（Tang和Wang，2018）利用卷积神经网络（CNN）捕捉序列模式和一般用户偏好。3）**SASRec**（Kang和McAuley，2018）是最具代表性的序列模型之一，它利用双向自注意力机制有效地建模复杂的用户行为。4）**TIGER**（Rajput等人，2023）是一种具有代表性的生成式推荐方法，它采用 RQ-VAE，并通过码本

[4] https://jmcauley.ucsd.edu/data/amazon/。 [5] https://github.com/westlake-repl/MicroLens/。 [6] https://cseweb.ucsd.edu/~jmcauley/datasets/goodreads.html/。

将每个物品表示为由一系列外部令牌组成的序列。5）**LETTER** （Wang等人，2024b）也是一种基于外部令牌的方法，它将语义信号和协同过滤信号整合到物品标识符中，从而实现能够捕捉物品协同相似性的推荐。6）**SETRec** （Lin等人，2025年）是一种近期提出的方法，引入了顺序无关的物品标识符，以便更好地建模用户在不同维度上的偏好。7）**D3** （Bao等人，2024年）使用人类词汇中的物品标题来表示每个物品，并利用协同信息强调更重要的词元，从而实现可靠的推荐。8）**LatentR**^{3} （Zhang等人，2025c）是一种近期提出的推理增强型大语言模型方法，其中大语言模型会先通过额外的注意力模块，在潜在空间中推理用户偏好，然后再生成下一个物品。我们还扩展了一种并行潜在推理方法：9）**LatentTTS** （You等人，2025b），该方法利用蒙特卡洛丢弃和加性高斯噪声，使潜在推理多样化，并选择置信度较高的路径。对于其他无法直接扩展到推荐任务的现有可信大语言模型推理方法，我们在附录7.1）中进行了详细说明。

4.1.3 实现细节。

对于传统方法，我们分别在以下范围内搜索学习率和权重衰减： $\{1e^{-3}, 1e^{-4}, 1e^{-5}\}$和 $\{1e^{-3}, 1e^{-4}, 1e^{-5}, 1e^{-6}\}$。对于基于大语言模型的基线方法和 VRec，我们均在 Qwen2.5-1.5B（通义千问等人，2025年）上实例化，并使用从$3e^{-4}$、 $5e^{-5}$、 $3e^{-5}$中搜索得到的学习率，对全部大语言模型参数进行微调。对于 LatentR^3 和 VRec，我们在 $\{1, 2, 4, 6, 8, 10\}$中搜索推理步骤 m。验证器预测损失 β 的系数和单调性正则化 γ 的强度分别从 $\{0.3, 0.5, 0.7, 1\}$和 $\{0.1, 0.3, 0.5, 0.7, 1, 1\}$中选取。所有实验均基于预训练的 LatentR^3 实现 VRec。所有实验均在八块 NVIDIA A100-SXM4-80G GPU 上进行。更多实现细节请参阅匿名链接[7]。

4.2 总体性能（RQ1）

性能　　表现1中展示了 VRec 和各基线方法的性能，我们可以观察到　　：

• 在基于大语言模型的生成模型中，使用人类词汇的方法（D3 和 LatentR3）优于基于码本的方法（TIGER 和 LETTER）或基于嵌入的方法（SETRec）。这是合理的，因为基于码本的方法使用了大语言模型预训练中未曾见过的外部令牌。因此，当扩展到具有十亿级参数的大语言模型时，这类方法通常需要大量数据，才能有效适配由外部令牌索引的物品。我们也在（Bao等人，2024年；Lin 等人，2025年）中观察到了类似现象。

• 在各基线方法中，LatentR^3 始终优于所有其他方法，展现了先推理后推荐范式的有效性。通过引入专用于推理的额外注意力模块并采用强化学习，LatentR^3 被引导生成有用的推理嵌入，从而实现准确推荐。值得注意的是，LatentR^3 仅在 1 个推理步骤时取得最佳性能，这表明推理步骤缺乏可扩展性（参见第2节中的分析）。这一观察也与（Zhang 等人，2025c）报告的发现一致。

• VRec 在所有数据集上均持续取得出色的性能。具体而言，VRec-1Step 在大多数情况下优于基线方法。这一结果在意料之中，因为该方法引入了验证步骤，验证器可以从外部评估并调整可能不准确的推理步骤。值得注意的是，扩展到多个步骤后，VRec 获得了最佳总体性能，展现出可验证推理范式的可扩展性和稳健性。关于推理可扩展性的进一步分析，以及有关同质性缓解的案例研究，分别见第 4.3.2 节和附录 7.2。

4.3 深入分析

4.3.1 消融研究（RQ2）

为了研究验证步骤在 VRec 中的有效性，我们移除了验证步骤，并将其记为“无验证器”。此外，为了验证多种验证器的有效性，我们将 VRec 实例化为仅包含类别维度或标题维度中的单一验证器，分别记为“单一 V-C”和“单一 V-T”。我们

[7] https://anonymous.4open.science/r/VRec/。

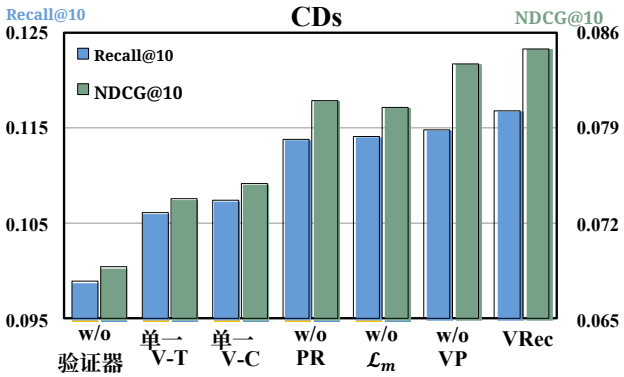


图 5 VRec在CDs数据集上的消融研究。

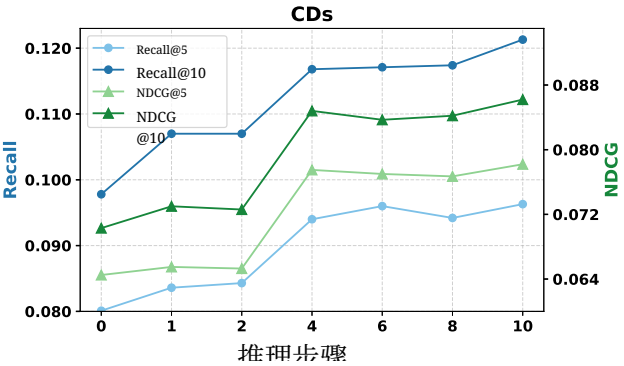


图 6 VRec推理步骤的可扩展性。

还分别移除了个性化路由器（“无 VP”）和单调性正则化（“无 $\mathcal{L}_m$”），并移除了验证器预训练阶段（“无 VP”），以评估这些组件各自的影响。

CDs数据集上的结果[8] 如图 5所示，我们可以发现：1）移除验证器会禁用验证步骤（即，LatentR^3），导致性能显著下降。这验证了引入验证器的必要性，并展现了推理-验证-推荐范式的优越性。此外，2）使用单一验证器会限制验证步骤的有效性，这凸显了多维验证器对于实现更准确验证的重要性。同时，缺少个性化路由器的VRec表现较差，进一步表明个体特定验证的重要性。此外，3）移除单调性正则化会损害性能。这一结果符合预期，因为该正则化可以惩罚导致偏好对齐较弱的推理。最后，4）直接联合训练验证器和大语言模型推荐器会导致推荐结果较差，这可能是因为如果验证器从头开始训练，验证器预测任务与下一物品预测任务之间可能存在冲突。

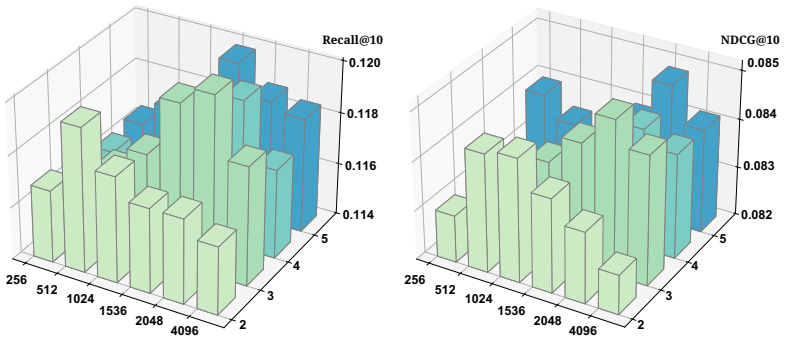
4.3.2 推理步骤可扩展性（RQ3）

为探究可验证推理范式的潜力，我们将完整推理步骤的数量 m从 0 调整到 10，并在图6中报告结果。从结果中可以观察到：1）随着推理步骤增加，推荐效果逐步提升。这凸显了对中间推理进行验证的重要性，因为验证有可能纠正与用户偏好不一致的推理过程。尽管 VRec 展现出良好的测试时可扩展性，2）性能提升趋于停滞（推理步骤从 6 增至 10）。这可能是因为单调性正则化会促使预测确定性不断提高，使验证器随着推理步骤的深入，越来越难以实现更准确的偏好对齐，从而限制了后续阶段的显著提升。

[8]为节省篇幅，我们省略了其他具有类似观察结果的数据集的结果。

表2 性能比较不同验证器维度下 VRec 在 CDs数据集上的表现。

	Recall@5	Recall@10	NDCG@5	NDCG@10
类别	0.0854	0.1074	0.0678	0.0749
类别 + 标题	0.0940	0.1168	0.0775	0.0848
类别 + 标题 +协同过滤	0.0973	0.1223	0.0796	0.0877



(a) 在 Recall@10 方面的性能 **(b) 在 NDCG@10 方面的性能**

图7 VRec 在不同验证器模型规模下于 CD 数据集上的性能比较 t.

4.3.3 验证器可扩展性（RQ3）

在 VRec 中，关键组件是验证器混合，其中验证器的模型容量也可能是实现有效可验证推理的重要因素。我们从两个角度考察验证器的可扩展性：

• 验证器维度。我们逐步增加验证器的数量，分别对应类别、标题和协作信号，以考察扩展验证维度的影响。表 2 的结果表明，随着评估维度的拓宽，性能持续提升。值得注意的是，语义维度带来的提升比协作特征更为显著，这可能是因为大语言模型的推理空间主要是语义性的，因此不太擅长捕捉协作模式。先前的研究也报告了类似的观察（Zhang 等人，2025年b）。

• 验证器模型规模。我们采用多层感知机（MLP）构建验证器，以探究其可扩展性。图 7 的结果表明，适度增加验证器的宽度或深度可以提升验证质量，从而带来更准确的推荐，例如 3 层、隐藏维度为 2048。不过，持续扩大验证器规模并无必要，因为它执行的是群组级预测任务，类别数量有限（约 20 个）。因此，过大的模型容量可能导致过拟合，而非进一步提升性能。

4.3.4 验证器效率分析（RQ4）

为实证评估验证步骤带来的计算开销及其在实际应用中的可行性，我们评估了 VRec 在不同推理步骤下启用和不启用验证时的表现。如表 3 所示，加入验证会带来适度的计算成本增加，这与预期一致。不过，由于大部分计算由大语言模型骨干网络主导，开销仍可忽略不计（平均额外耗时为 0.59%）。因此，可以灵活调整推理步骤，以平衡推荐性能与推理效率，同时仍能受益于验证带来的性能提升。

4.3.5 超参数敏感性（RQ4）

我们进一步分析 VRec 的超参数，以促进其未来应用。1）**验证器损失强度的影响**β。从图8(a) 可以观察到，随着 β从 0 变化到 1，推荐准确率先上升后下降。验证器需要保持足够的预测能力，才能提供有效的引导，

表3 CDs数据集上有验证器（“w/”）与无验证器（“w/o”）的大语言模型推理平均推理耗时（每个样本）对比。

# 推理步骤	1	2	4	6	8	10
无验证器	0.754s	0.825s	0.896s	0.946秒	1.064秒	1.142秒
使用验证器	0.761秒	0.831秒	0.908s	0.950s	1.065s	1.142s
% 时间开销	0.97%	0.71%	1.33%	0.46%	0.06%	0.03%

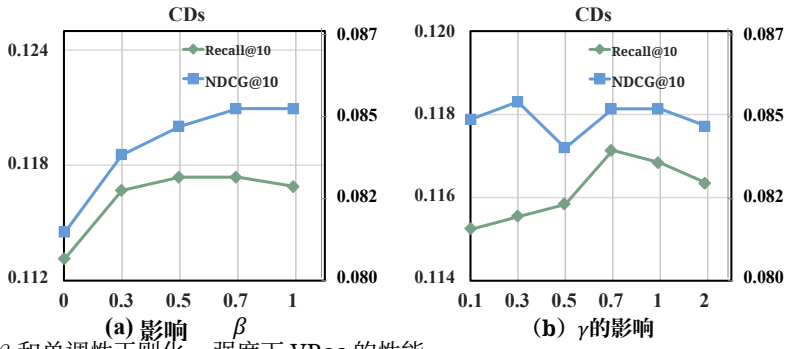


图 8 不同验证器损失 β 和单调性正则化 γ强度下 VRec 的性能。

但过强的验证器可能会与推荐目标冲突。2）**单调性正则化强度的影响** γ。如图8(b) 所示，增强单调性正则化可持续提升性能，证明其有助于促进稳定的推理细化。当 γ足够大时，性能提升会逐渐趋于饱和，且不会导致性能下降。这是合理的，因为正则化项仅惩罚非单调的熵变化。

4.3.6 关于可解释性的讨论。

尽管在潜在空间中进行推理非常有效且高效，但解释推理与验证过程十分重要，也颇具挑战性。值得庆幸的是，我们提出的 VRec 有望借助验证器混合来解释推理过程。它可以利用验证器的群组级预测来解释推理过程（从“爵士音乐”到“钢琴爵士乐”）。然而，严格评估这类解释与现实世界中的用户决策之间的一致性并非易事，因此我们将这项研究留待未来开展。

5 相关工作

5.1 推荐推理

近期进展凸显了将推理融入推荐模型的潜力（Liu等人，2025年；Dai等人，2025年），即模型在给出最终推荐之前会执行多个中间推理步骤（Wang等人，2024年c；Zhao等人，2025年；Xie等人，2025年；Gu等人，2025年；Xing等人，2025年；Huang等人，2025年）。在基于大语言模型的推荐中，现有工作大致可分为显式推理和隐式推理。显式方法通常依赖监督式思维链（CoT）推理，且主要用于点击率或评分预测，而非下一项生成（Xi等人，2024年；Wu等人，2025年；Fang等人，2025年）。然而，构建思维链数据成本高昂，显式推理也常常效率低下，这给大规模大语言模型部署带来了重大挑战。相比之下，隐式（潜在）推理无需标注的思维链数据，效率也更高（Tang等人，2025年b；Lin等人，2026年）。尽管如此，先前研究不可避免地会遇到未经验证的推理，导致同质化捷径推理以及步骤间的误差累积，从而限制基于推理的推荐的效果。为弥合这一差距，我们提出一种新颖的可验证推理范式，在每个推理步骤提供中间反馈，审查推理过程，使其更忠实地理解用户偏好，并释放推理的潜力。

5.2 可验证的大语言模型推理

可验证的大语言模型推理要求对模型的推理过程进行结构化，使每个中间步骤都能接受正确性检查。现有研究大致可分为两类。一类工作探索外部验证器，利用专门模块或奖励模型显式评估中间推理步骤（Wang等人，2024年a；Zhang等人，2025年a、d；Zheng等人，2025年a）。Self-Refine（Madaan等人，2023年）、CRITIC（Wang等人，2023年b；Zheng等人，2025年b）和 Tree-of-Thoughts（Yao等人，2023年）等框架通过学习得到的评判器或基于搜索的评估器，迭代验证并优化推理轨迹。相比之下，第二类内部信号方法从模型自身的输出中提取可靠性线索。自一致性（Wang等人，2023年a；Li等人，2023年；Taubenfeld等人，2025年）汇总多个推理样本以识别一致的结果，而基于置信度和熵的度量（Wang等人，2025年）则量化不确定性并进行校准。然而，这些方法依赖显式推理轨迹（例如文本思维链），因此效率低下，也难以推广到下一项推荐中的潜在推理场景。在本工作中，我们针对基于大语言模型的推荐提出了一个通用的可验证推理框架，该框架以潜在推理嵌入为基础，实现逐步推理验证。

6 结论

在这项工作中，我们指出了现有推理增强型基于大语言模型的推荐中的一个关键问题，即未经验证的推理可能导致推理退化。随后，我们提出了一种新颖的「推理—验证—推荐」范式，通过引入额外的验证步骤，为推理过程提供可靠的反馈引导。接着，我们总结了验证器设计的两项原则，即：1）可靠的评估与引导反馈；2）多维验证。为满足这两项原则，我们提出了 VRec，它采用由个性化路由器控制的验证器混合，以确保多维验证。为实现可靠的验证，我们为验证器制定了群体级偏好预测任务，并利用模型内部信号（即熵和权重）进行验证。实证结果表明，VRec 具有有效性、高效率和可扩展性。

本研究初步探索了下一项推荐中大语言模型推理的验证问题，仍有若干值得探索的方向。1）将我们的框架扩展到其他推荐任务中的显式推理，例如点击率或评分预测，值得进一步研究。2）尽管可验证推理在潜在空间中效率较高，但提升其可解释性和可控性也至关重要。第三，验证器设计蕴含着丰富的探索机会，例如开发表达能力更强的架构和更优的对齐目标，以强化偏好建模和与人类对齐的验证。

7 附录

7.1 详细描述

- **De对同质化推理问题的详细分析。**根据优化目标，即 .,

$$\arg\min_{\theta\in\Theta} \mathbb{E}_{X\sim\mathcal{D}}\, \mathbb{E}_{R\sim\mathcal{M}_\theta(\cdot|X)}\, \mathcal{L}_\theta(\hat{y} \mid X, R), \tag{15}$$

我们可以观察到，监督信号仅作用于最终推荐 ˆy，而中间推理过程 R 并未得到直接优化。因此，推理模型 $\mathcal{M}_\theta(R \mid X)$ 可能会收敛到一种退化分布 ，并坍缩为单一的确定性推理模式：

$$\mathcal{M}_\theta(R \mid X) \to \delta_{R^*(X)}(R), \tag{16}$$

其中，$\delta_{R^*(X)}(R)$ 表示以 $R^*(X)$ 为中心的狄拉克δ分布，而 $R^*(X) = \arg\max_R \mathcal{M}_\theta(R \mid X)$ 表示给定输入 X 时最可能（或最优）的推理序列。换言之，模型倾向于生成同质或简单的推理步骤，缺乏针对不同输入的多样性。

- **VRec 的训练与流程流程。**我们在算法1中展示了 VRec 实例化的整体流程，包括两阶段训练和推理。

• **群组级用户偏好标注**。为了构建偏好预测任务的监督信号，我们为每个物品分配一个群组级标签 P，以反映粗粒度的用户偏好。具体而言，我们在这项工作中利用三类信息：1）类别。如果数据集提供了带有预定义类别（即 “爵士乐”或“嘻哈音乐”）的物品元数据，我们就直接将这些类别用作群组标签；2）标题语义。我们从物品标题中提取语义嵌入，并进行 k均值聚类，将文本内容相似的物品分组；3）协同信息。基于预训练的协同过滤（CF）模型（如 SASRec）Kang和McAuley (2018)，我们使用 k均值聚类对物品协同过滤嵌入进行聚类，以捕捉协同相似性。所得群组分配作为偏好标签 P，用于训练偏好预测任务中的验证器。

算法1 VRec的训练与推理流程

输入： 预训练的推理型大语言模型推荐器参数 θ、初始化的验证器参数 ϕ、训练数据集 $\mathcal{D}$，推理步骤总数 m、验证器损失系数 β以及单调性正则化系数 γ。

// 第一阶段：验证器预训练

对于所有 $(X, y) \in \mathcal{D}$ **执行**

  使用 θ生成推理 R 和推荐$\hat{y}$；　▷ 式（1）- (2)

  $\mathcal{D}_v \leftarrow \varnothing$

  如果 $\hat{y} = y$ **那么**

    $\mathcal{D}_v \leftarrow \mathcal{D}_v \cup (R, \mathcal{P})$, $\mathcal{P} = \{P_1, \ldots, P_n\}$; // 正样本

  else

    $\mathcal{D}_v \leftarrow \mathcal{D}_v \cup (R, \varnothing)$ // 负样本

使用 $\mathcal{L}_v$ 优化验证器 ϕ，针对 $\mathcal{D}_v$;

// 第二阶段：可验证推理微调

对于所有 $(X, y) \in \mathcal{D}$ **执行**

  对于 $t = 1$ 到 m **执行**

    从大语言模型推荐器获取 $\boldsymbol{r}_t$；　▷ 公式（1）

    通过公式（$\{f_i, \boldsymbol{g}_i\}_{i=1}^{n}$ 8）获取;　▷ 多维评估

    获取 $\boldsymbol{r}_t^*$ 通过式（10）　▷ 基于置信度的调整

    更新 $R \leftarrow (R, \boldsymbol{r}_t^*)$;

    计算 $\mathcal{L}_m = \max(0, f_i^t - f_i^{t-1})$ 通过式（13）；

  计算推荐损失 $\mathcal{L}_r$;　▷ Eq.(12)

  更新 θ 和 ϕ，联合采用：$\mathcal{L} = \mathcal{L}_r + \beta\mathcal{L}_v + \gamma\mathcal{L}_m$.　▷ Eq.(14)

// 推理

对 $t = 1$ 到 m **执行**

  执行推理步骤以获得 $\boldsymbol{r}_t$;

  $\boldsymbol{r}_t^* \leftarrow \boldsymbol{r}_t$ 通过公式(10);　▷基于置信度的调整

生成最终推荐$\hat{y}$;　▷ 公式 (2)

输出： 最终训练参数 $\{\theta, \phi\}$ 和推荐$\hat{y}$。

每个维度的类别数 d_i。关于每个验证维度的类别数：

• 对于类别方面，我们采用最粗粒度物品类别的数量，该数量在数据集中约为20。具体而言，我们利用CD（27个类别）和乐器（19个类别）数据集中的类别维度进行验证。

• 对于标题维度和协同维度，我们通过实证为每个数据集从 $\{15, 20, 25\}$ 中选取一个预定义数量。其范围与类别维度保持一致，旨在平衡不同维度下预测任务的难度。

表4 数据集统计信息。“交互”表示“交互次数”。

	# 训练交互.	# 验证交互	# 测试交互	# 物品
CDs	49,251	6,156	6,158	5,841
MicroLens	60,101	9,036	9,121	5,326
Goodreads	256,868	33,148	33,040	7,413
乐器	66,500	8,312	8,313	5,030

• **详细实现与设置**。以下是 VRec 中验证器的其他实现细节和实验设置。

验证器数量设置。 为确保不同数据集上的验证器配置保持一致，我们使用两个验证器进行总体性能比较（第 4.2 节）。值得注意的是，*MicroLens* 和 *Goodreads* 数据集不包含结构化物品类别信息，因此可用的群体级偏好维度数量有限。为进一步研究扩展验证器维度能否提升性能，我们在第4.3.3 节进行了额外分析。结果表明，纳入更多样化的验证器维度能够更有效地进行推理，并持续提升性能。

验证器模型规模设置。 为深入分析验证器的可扩展性（第 4.3.3 节），我们使用多层感知机实现验证器，并分别从$\{256, 512, 1024, 1536, 2048, 4096\}$ 和 $\{2, 3, 4, 5\}$的范围内选择隐藏维度和隐藏层数。对于所有其他实验，我们仅使用线性矩阵 $W_v \in \mathbb{R}^{d_\mathcal{M} \times d_i}$实现验证器。

推理步骤设置。 在深入分析实验中，我们发现，随着推理步骤数从 1 增加到 10，VRec 的性能持续提升，之后逐渐趋于饱和。由于增加推理步骤数不可避免地会提高测试时计算成本，我们采用了一种兼顾性能与计算效率的配置。具体而言，在*Amazon CD* 数据集上将推理步骤数设为 4，在*MicroLens* 数据集上设为 6，从而在推理效果与推理效率之间取得了较好的平衡。

7.2 补充实验结果

• **案例研究**。为分析 VRec 如何缓解同质化问题，我们通过 t-SNE 可视化展示推理表示、引导向量和目标物品表示，如图 9 所示。可视化结果清楚地展现了从初始推理表示逐渐趋向目标物品表示的引导趋势。这验证了 VRec 在缓解推理退化问题方面的有效性，从而带来更恰当的推荐。

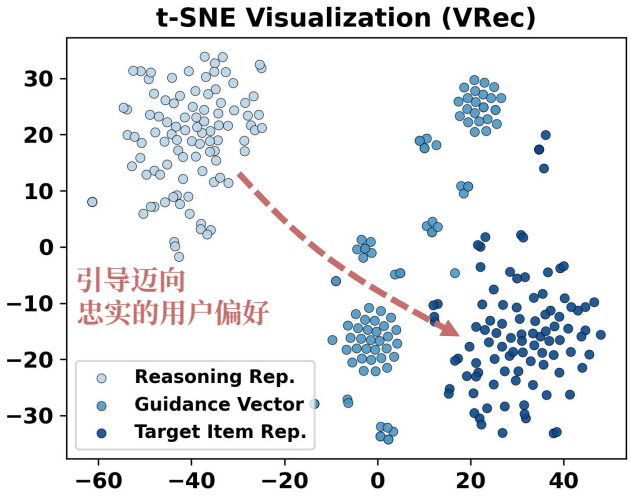


图 9 VRec 在 CD 上逐步提升推理准确性的可视化结果。

• **MicroLens 上的消融研究。**消融研究结果如图所示10，由此我们得到与 CD 上类似的观察结果，包括验证器的影响最为显著（“无验证器”的性能最差），以及 VRec 中各组件的有效性。尽管如此，我们还发现了另一个有趣的现象：验证协同维度的推理（“单一 V-CF”）比验证语义维度的推理（“单一 V-T”）更有帮助。一个可能的原因是，MicroLens 中的文本描述存在噪声，例如，包含与微视频主要内容无关的超长句子。在这种情况下，协同信息有助于表征一些重要的物品特征，例如流行度。

• **MicroLens数据集上的推理步骤可扩展性。**图中展示了 VRec 在不同推理步骤下的性能 m在 MicroLens 数据集上的结果，如图11所示。从结果中，我们得出了与

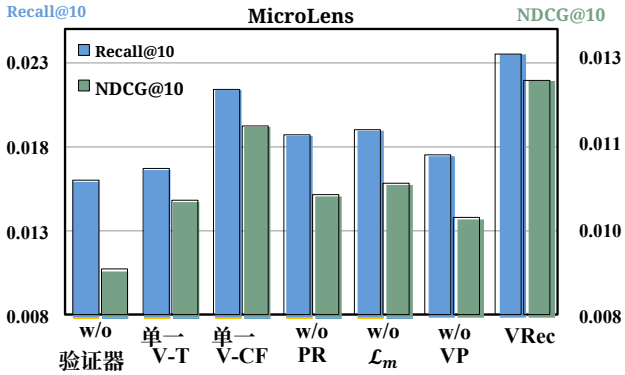


图10 推理步骤的可扩展性。

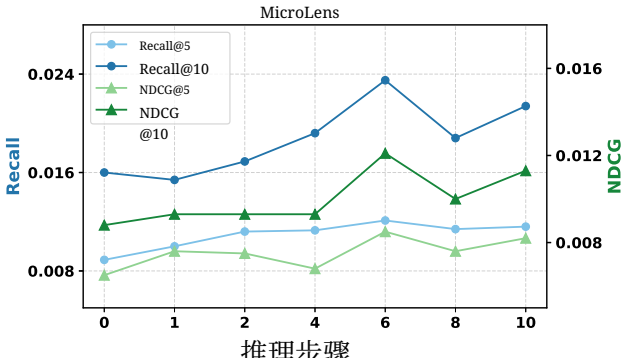


图11 推理步骤的可扩展性。

CDs数据集上相似的观察结果，这证明了 VRec 在缓解推理退化问题方面的优势，从而使测试时推理步数具有良好的可扩展性。

参考文献

Keqin Bao、Jizhi Zhang、Yang Zhang、Xinyue Huo、Chong Chen 和 Fuli Feng。解码至关重要：解决大语言模型推荐中的放大偏差和同质性问题。载于*ACL*。ACL，2024年。

Keqin Bao、Jizhi Zhang、Wenjie Wang、Yang Zhang、Zhengyi Yang、Yanchen Luo、Chong Chen、Fuli Feng 和 Qi Tian。面向推荐系统中大语言模型的双步接地范式。*ACM*《推荐系统汇刊》，3(4):1–27，2025年。

Sunhao Dai、Jiakai Tang、Jiahua Wu、Kun Wang、Yuxuan Zhu、Bingjun Chen、Bangyang Hong、Yu Zhao、Cong Fu、Kangle Wu 等。Onepiece：将上下文工程和推理引入工业级级联排序系统。*arXiv* 预印本 *arXiv:2509.18091*，2025年。

Hao Ding、Ziwei Fan、Ingo Guehring、Gaurav Gupta、Wooseok Ha、Jun Huan、Linbo Liu、Behrooz Omidvar-Tehrani、Shiqi Wang 和 Hao Zhou。大语言模型在代码开发中的推理与规划。发表于*KDD*，第6480–6490页，2024a。

Mengru Ding、Hanmeng Liu、Zhizhang Fu、Jian Song、Wenbo Xie 和 Yue Zhang。打破思维链：大语言模型可能是捷径推理者。*arXiv:2406.06580*，2024年b。

Yi Fang、Wenjie Wang、Yang Zhang、Fengbin Zhu、Qifan Wang、Fuli Feng 和 Xiangnan He。Reason4Rec：借助审慎式用户偏好对齐实现推荐的大语言模型。*arXiv*预印本 *arXiv:2502 02061*，2025年。。

Sebastian Farquhar、Jannik Kossen、Lorenz Kuhn 和 Yarin Gal。使用语义熵检测大语言模型中的幻觉。《自然》，630(8017)：625–630，2024年。

Yichao Fu、Xuewei Wang、Yuandong Tian 和 Jiawei Zhao。有信心地深度思考。*arXiv:2508.15260*，2025年。

Robert Geirhos、Jörn-Henrik Jacobsen、Claudio Michaelis、Richard Zemel、Wieland Brendel、Matthias Bethge 和 Felix A Wichmann。深度神经网络中的捷径学习。Nature Machine Intelligence，2(11):665–673，2020。Hao Gu、Rui Zhong、Yu Xia、Wei Yang、Chi Lu、Peng Jiang 和 Kun Gai。R4ec：面向推荐系统的推理、反思与精炼框架，2025年。https://arxiv.org/abs/2507.17249。Daya Guo、Dejian Yang、Haowei Zhang、Junxiao Song、Ruoyu Zhang、Runxin Xu、Qihao Zhu、Shirong Ma、Peiyi Wang、Xiao Bi 等。Deepseek-r1 12948，2025年 2501：通过强化学习激励大语言模型的推理能力。arXiv: . 。Balázs Hidasi、Alexandros Karatzoglou、Linas Baltrunas 和 Domonkos Tikk。使用循环神经网络的基于会话的推荐。载于 ICLR，2016。Chengkai Huang、Junda Wu、Yu Xia、Zixu Yu、Ruhan Wang、Tong Yu、Ruiyi Zhang、Ryan A. Rossi、Branislav Kveton、Dongruo Zhou、Julian McAuley 和 Lina Yao。迈向多模态大语言模型时代的智能体推荐系统，2025年。https://arxiv.org/abs/2503.16734。Yichen Huang 和 Lin F. Yang。借助模型无关的验证与精炼流程在2025年国际数学奥林匹克中夺金，2025年。Wang-Cheng Kang 和 Julian McAuley。自注意力序列推荐。载于 ICDM，第197–206页。IEEE，2018。Jieyong Kim、Hyunseo Kim、Hyunjin Cho、SeongKu Kang、Buru Chang、Jinyoung Yeo 和 Dongha Lee。面向推荐的基于大语言模型的评论驱动的个性化偏好推理。corr，abs/2408.06276，2024年。doi: 10.48550。arXiv preprint ARXIV.2408.06276。Oscar Li、Hao Liu、Chaofan Chen 和 Cynthia Rudin。通过原型实现基于案例的推理的深度学习：一种能够解释其预测的神经网络。载于 AAAI，第32卷，2018。Yifei Li、Zeqi Lin、Shizhuo Zhang、Qiang Fu、Bei Chen、Jian-Guang Lou 和 Weizhu Chen。借助步骤感知验证器使大语言模型成为更好的推理者，2023年。https://arxiv.org/abs/2206.02336。Xinyu Lin、Wenjie Wang、Yongqi Li、Fuli Feng、See-Kiong Ng 和 Tat-Seng Chua。连接物品与语言：面向基于大语言模型的推荐的过渡范式。载于 KDD，第1816–1826页。ACM，2024年。Xinyu Lin、Haihan Shi、Wenjie Wang、Fuli Feng、Qifan Wang、See-Kiong Ng 和 Tat-Seng Chua。面向基于大语言模型的生成式推荐的无序标识符。载于 SIGIR，第1923–1933页，2025年。Xinyu Lin、Pengyuan Liu、Wenjie Wang、Yicheng Hu、Chen Xu、Fuli Feng、Qifan Wang 和 Tat-Seng Chua。从级联排序的视角将推理引入生成式推荐。arXiv:2602.03692，2026。Fei Liu、Xinyu Lin、Hanchao Yu、Mingyuan Wu、Jianyu Wang、Qiang Zhang、Zhuokai Zhao、Yinglong Xia、Yao Zhang、Weiwei Li 等。Recoworld：为智能体推荐系统构建模拟环境。载于 arXiv:2509 10397，2025年 . 。Hao Liu、Zi-Yi Dou、Yixin Wang、Nanyun Peng 和 Yisong Yue。面向工具使用型语言智能体的不确定性校准。载于 EMNLP，第16781–16805页，2024年。Aman Madaan、Niket Tandon、Prakhar Gupta、Skyler Hallinan、Luyu Gao、Sarah Wiegreffe、Uri Alon、Nouha Dziri、Shrimai Prabhumoye、Yiming Yang、Shashank Gupta、Bodhisattwa Prasad Majumder、Katherine Hermann、Sean Welleck、Amir Yazdanbakhsh 和 Peter Clark。Self-refine：基于自我反馈的迭代精炼，2023年。https://arxiv.org/abs/2303.17651。Frederic Morin 和 Yoshua Bengio。分层概率神经网络语言模型。载于人工智能与统计国际研讨会，第246–252页。PMLR，2005。Yongxin Ni、Yu Cheng、Xiangyan Liu、Junchen Fu、Youhua Li、Xiangnan He、Yongfeng Zhang 和 Fajie Yuan。大规模内容驱动的微视频推荐数据集。arXiv:2309.15379，2023年。OpenAI。学习借助大语言模型进行推理，2025年。https://openai.com/index/learning-to-reason-with-llms/。

通义千问：An Yang、Baosong Yang、Beichen Zhang、Binyuan Hui、Bo Zheng、Bowen Yu、Chengyuan Li、Dayiheng Liu、Fei Huang、Haoran Wei、Huan Lin、Jian Yang、Jianhong Tu、Jianwei Zhang、Jianxin Yang、Jiaxi Yang、Jingren Zhou、Junyang Lin、Kai Dang、Keming Lu、Keqin Bao、Kexin Yang、Le Yu、Mei Li、Mingfeng Xue、Pei Zhang、Qin Zhu、Rui Men、Runji Lin、Tianhao Li、Tianyi Tang、Tingyu Xia、Xingzhang Ren、Xuancheng Ren、Yang Fan、Yang Su、Yichang Zhang、Yu Wan、Yuqiong Liu、Zeyu Cui、Zhenru Zhang 和 Zihan Qiu。Qwen2.5 技术报告。2025年。

Shashank Rajput、Nikhil Mehta、Anima Singh、Raghunandan H Keshavan、Trung Vu、Lukasz Heldt、Lichan Hong、Yi Tay、Vinh Q Tran、Jonah Samost 等。采用生成式检索的推荐系统。载于 NeurIPS。Curran Associates, Inc.，2023年。Jiakai Tang、Sunhao Dai、Teng Shi、Jun Xu、Xu Chen、Wen Chen、Jian Wu 和 Yuning Jiang。推荐前先思考：释放序列推荐的潜在推理能力。arXiv 预印本 arXiv:2503.22675，2025a年。Jiakai Tang、Sunhao Dai、Teng Shi、Jun Xu、Xu Chen、Wen Chen、Jian Wu 和 Yuning Jiang。推荐前先思考：释放序列推荐的潜在推理能力，2025b年。https://arxiv.org/abs/2503.22675。Jiaxi Tang 和 Ke Wang。通过卷积序列嵌入实现个性化 Top-N 序列推荐。载于 WSDM，第565–573页，2018年。Amir Taubenfeld、Tom Sheffer、Eran Ofek、Amir Feder、Ariel Goldstein、Zorik Gekhman 和 Gal Yona。置信度提升大语言模型中的自一致性。载于计算语言学协会论文集：ACL 2025 Findings，第20090–20111页。计算语言学协会，2025年。DOI：10.18653/v1/2025.findings-acl.1030。http://dx.doi.org/10.18653/v1/2025.findings-acl.1030。Kimi Team、Yifan Bai、Yiping Bao、Guanduo Chen、Jiahao Chen、Ningxin Chen、Ruijue Chen、Yanru Chen、Yuankun Chen、Yutian Chen 等。Kimi k2：开放式智能体智能。arXiv:2507.20534，2025年。Qwen Team 等。Qwen2 技术报告。arXiv 预印本 arXiv:2407.10671，2:3，2024年。Alicia Y Tsai、Adam Kraft、Long Jin、Chenwei Cai、Anahita Hosseini、Taibai Xu、Zemin Zhang、Lichan Hong、Ed H Chi 和 Xinyang Yi。利用大语言模型推理增强个性化推荐系统。arXiv 预印本 arXiv:2408.00802，2024年。Peiyi Wang、Lei Li、Zhihong Shao、R. X. Xu、Damai Dai、Yifei Li、Deli Chen、Y. Wu 和 Zhifang Sui。Math-shepherd：无需人工标注，逐步验证并强化大语言模型，2024a年。https://arxiv.org/abs/2312.08935。Shenzhi Wang、Le Yu、Chang Gao、Chujie Zheng、Shixuan Liu、Rui Lu、Kai Dang、Xionghui Chen、Jianxin Yang、Zhenru Zhang、Yuqiong Liu、An Yang、Andrew Zhao、Yang Yue、Shiji Song、Bowen Yu、Gao Huang 和 Junyang Lin。超越二八法则：高熵少数词元推动有效的大语言模型推理强化学习，2025年。https://arxiv.org/abs/2506.01939。Wenjie Wang、Fuli Feng、Xiangnan He、Liqiang Nie 和 Tat-Seng Chua。为推荐去噪隐式反馈。载于 WSDM，第373–381页。ACM，2021年。Wenjie Wang、Honghui Bao、Xinyu Lin、Jizhi Zhang、Yongqi Li、Fuli Feng、See-Kiong Ng 和 Tat-Seng Chua。可学习的物品词元化用于生成式推荐。载于 CIKM，第2400–2409页，2024b年。Xuezhi Wang、Jason Wei、Dale Schuurmans、Quoc Le、Ed Chi、Sharan Narang、Aakanksha Chowdhery 和 Denny Zhou。自一致性提升语言模型中的思维链推理，2023a年。https://arxiv.org/abs/2203.11171。Yan Wang、Zhixuan Chu、Xin Ouyang、Simeng Wang、Hongyan Hao、Yue Shen、Jinjie Gu、Siqiao Xue、James Y Zhang、Qing Cui、Longfei Li、Jun Zhou 和 Sheng Li。利用大语言模型推理图增强推荐系统，2024c年。https://arxiv.org/abs/2308.10835。Yu Wang、Zhiwei Liu、Jianguo Zhang、Weiran Yao、Shelby Heinecke 和 Philip S. Yu。DRDT：用于大语言模型序列推荐的发散思维动态反思，2023b年。https://arxiv.org/abs/2312.11336。Hui Wei、Zihao Zhang、Shenghua He、Tian Xia、Shijia Pan 和 Fei Liu。PlanGenLLMs：大语言模型规划能力的现代综述。ACL，2025年。Jason Wei、Xuezhi Wang、Dale Schuurmans、Maarten Bosma、Fei Xia、Ed Chi、Quoc V Le、Denny Zhou 等。思维链提示引发大语言模型中的推理。第35卷，第24824–24837页，2022年。Chenghao Wu、Ruiyang Ren、Junjie Zhang、Ruirui Wang、Zhongrui Ma、Qi Ye 和 Wayne Xin Zhao。STARec：通过自主审慎推理实现的高效推荐系统智能体框架。载于第34届 ACM 信息与知识管理国际会议论文集（CIKM），2025年。https: //arxiv.org/abs/2508.18812。Yunjia Xi、Weiwen Liu、Jianghao Lin、Xiaoling Cai、Hong Zhu、Jieming Zhu、Bo Chen、Ruiming Tang、Weinan Zhang 和 Yong Yu。借助大语言模型知识增强，迈向开放世界推荐。载于 RecSys，第12–22页，2024年。Yu Xie、Xingkai Ren、Ying Qi、Yao Hu 和 Lianlei Shan。RecLLM-R1：结合强化学习与思维链的两阶段训练范式 v1，2025年。https://arxiv.org/abs/2506.19235。

Haibo Xing、Hao Deng、Yucheng Mao、Jinxin Hu、Yi Xu、Hao Zhang、Jiahao Wang、Shizhun Wang、Yu Zhang、Xiaoyi Zeng 和 Jing Zhang。Reg4rec：面向大规模推荐系统的推理增强生成模型，2025年。https://arxiv.org/abs/2508.15308。Shunyu Yao、Dian Yu、Jeffrey Zhao、Izhak Shafran、Thomas L. Griffiths、Yuan Cao 和 Karthik Narasimhan。思维树：借助大语言模型进行审慎的问题求解，2023年。https://arxiv.org/abs/2305.10601。Runyang You、Yongqi Li、Xinyu Lin、Xin Zhang、Wenjie Wang、Wenjie Li 和 Liqiang Nie。R^{2}ec：迈向具备推理能力的大型推荐模型。发表于 *NeurIPS*，2025a。Runyang You、Yongqi Li、Meng Liu、Wenjie Wang、Liqiang Nie 和 Wenjie Li。潜在推理模型的并行测试时扩展。 *arXiv:2510.07745*，2025b。Yu Yuan、Lili Zhao、Kai Zhang、Guangting Zheng 和 Qi Liu。大语言模型是否克服了捷径学习？对大语言模型中捷径挑战的评估。发表于 *EMNLP*，第12188–12200页，2024年。Lunjun Zhang、Arian Hosseini、Hritik Bansal、Mehran Kazemi、Aviral Kumar 和 Rishabh Agarwal。生成式验证器：将奖励建模作为下一个词元预测，2025a年。 https://arxiv.org/abs/2408.15240。Yang Zhang、Fuli Feng、Jizhi Zhang、Keqin Bao、Qifan Wang 和 Xiangnan He。Collm：将协同嵌入整合到大语言模型中用于推荐。 *TKDE*，2025b。Yang Zhang、Wenxin Xu、Xiaoyan Zhao、Wenjie Wang、Fuli Feng、Xiangnan He 和 Tat-Seng Chua。面向基于大语言模型的推荐的强化潜在推理。 *arXiv*预印本 *arXiv:2505.19092*，2025c。Zhenru Zhang、Chujie Zheng、Yangzhen Wu、Beichen Zhang、Runji Lin、Bowen Yu、Dayiheng Liu、Jingren Zhou 和 Junyang Lin。数学推理中过程奖励模型开发的经验教训，2025d年。 https: //arxiv.org/abs/2501.07301。Keyu Zhao、Fengli Xu 和 Yong Li。推理以推荐：运用交互式思维推理增强大语言模型推荐，2025年。 https://arxiv.org/abs/2506.05069。Chujie Zheng、Zhenru Zhang、Beichen Zhang、Runji Lin、Keming Lu、Bowen Yu、Dayiheng Liu、Jingren Zhou 和 Junyang Lin。Processbench：识别数学推理中的过程错误，2025a年。 https://arxiv.org/abs/ 2412.06559。Xin Zheng、Jie Lou、Boxi Cao、Xueru Wen、Yuqiu Ji、Hongyu Lin、Yaojie Lu、Xianpei Han、Debing Zhang 和 Le Sun。Critic-cot：通过思维链评论器提升大语言模型的推理能力，2025b年。 https://arxiv.org/abs/2408.16326。Xiong Zhou、Xianming Liu、Deming Zhai、Junjun Jiang、Xin Gao 和 Xiangyang Ji。面向不完善标注数据学习的原型锚定学习。发表于 *ICML*，第27245–27267页。PMLR，2022年。Zhanke Zhou、Rong Tao、Jianing Zhu、Yiwen Luo、Zengmao Wang 和 Bo Han。在含噪推理依据下进行思维链提示时，语言模型能否实现稳健推理？ *NeurIPS*，37：123846–123910，2024年。Yaochen Zhu、Liang Wu、Qi Guo、Liangjie Hong 和 Jundong Li。面向推荐系统的协同大语言模型。发表于 *WWW*，第3162–3172页，2024年。