

【LLM论文阅读】LLM4Rerank-多维度重排框架

原创 方方 方方的算法花园 2024年10月29日 14:45 北京

论文标题：

LLM-enhanced Reranking in Recommender Systems

论文链接：<https://arxiv.org/pdf/2406.12433>

论文作者所在机构：香港大学、华为诺亚方舟实验室

一句话概括：基于LLM的 LLM4Rerank 框架用于推荐系统重排序，通过构建包含多种节点的函数图结构及自动重排序过程，解决现有模型多方面平衡、可扩展性和个性化不足问题，经实验验证在多方面表现优于基线模型。

论文出发点

1. 现有重排序模型的局限性

(1) **多方面平衡困难**：现有重排序模型大多专注于推荐的准确性，虽准确性至关重要，但在实际应用中，推荐列表的多样性和公平性同样不容忽视。然而，由于这些方面在语义上存在较大差距，且各自通过独特的属性维度审视推荐列表，导致现有模型难以在建模时全面考虑并平衡这些复杂的多方面组合。

(2) **可扩展性问题**：不同的推荐场景可能优先考虑不同的方面或功能规则，尤其是在引入新的方面或自定义重排序规则（如反向规则或停止条件）时，现有模型的复杂性使其难以适应多样的推荐设置，可扩展性较差。

(3) **个性化不足**：现有模型在不同方面的输出倾向固定，无法根据不断变化的业务或用户偏好进行智能调整，缺乏个性化。

2. 大语言模型（LLM）在重排序中的潜力与挑战

(1) **LLM 在重排序任务中的表现**：尽管零样本 LLM 在信息提取和推荐等任务中可能不如专门模型，但在涉及有限数量项目的简短上下文中，其语义理解能力使其在重排序任务中能够表现出与监督基准相当甚至更优的性能。

(2) **应用 LLM 面临的挑战**：将 LLM 应用于重排序框架建模时，面临确保框架可扩展性以及根据特定推荐设置或用户偏好自动组合不同方面要求以实现个性化的挑战。

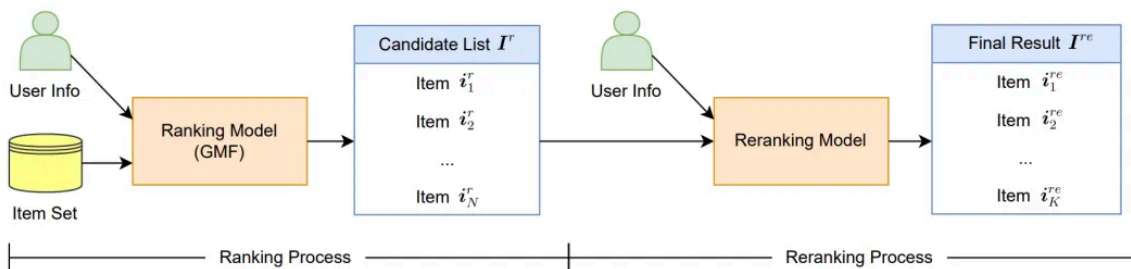


Figure 1: Ranking and reranking process in recommendations.

论文贡献点

1. 多方面整合与统一测量的首创尝试

据作者所知，本文首次实现自动整合多个方面，并通过采用 LLM 的多跳重排序程序，在统一语义空间中全面衡量不同方面。这一创新为推荐系统中的重排序任务提供了新的思路，使得在考虑准确性的同时，能够更好地兼顾多样性和公平性等多方面因素，突破了以往模型仅关注单一或少量方面的局限，从而更全面地优化推荐结果。

2.提出 LLM4Rerank 框架

提出了一种新颖的 LLM4Rerank 框架，该框架能够处理重排序过程中如准确性、多样性和公平性等各种复杂的方面要求组合。其优势在于通过特定的结构设计，如将各方面要求表示为不同节点，利用 Chain-of-Thought (CoT) 方式实现自动整合，确保了框架的可扩展性，使得能够灵活应对新的方面需求。同时，借助 LLM 根据当前重排序历史和用户提供的“Goal”自动确定下一步节点，实现了重排序过程的个性化，为用户提供更贴合其需求的推荐结果，在性能、可扩展性和个性化方面展现出巨大潜力。

3. 实验验证框架优势

在三个广泛使用的工业数据集上进行的实验表明，LLM4Rerank 在所有考虑的方面均优于现有的基线模型。这充分验证了该框架在增强推荐系统重排序过程中的性能、可扩展性和个性化方面的有效性和优越性，为其在实际推荐系统中的应用提供了有力的依据，证明了该框架能够在不同数据集和实际场景中发挥积极作用，显著提升推荐质量。

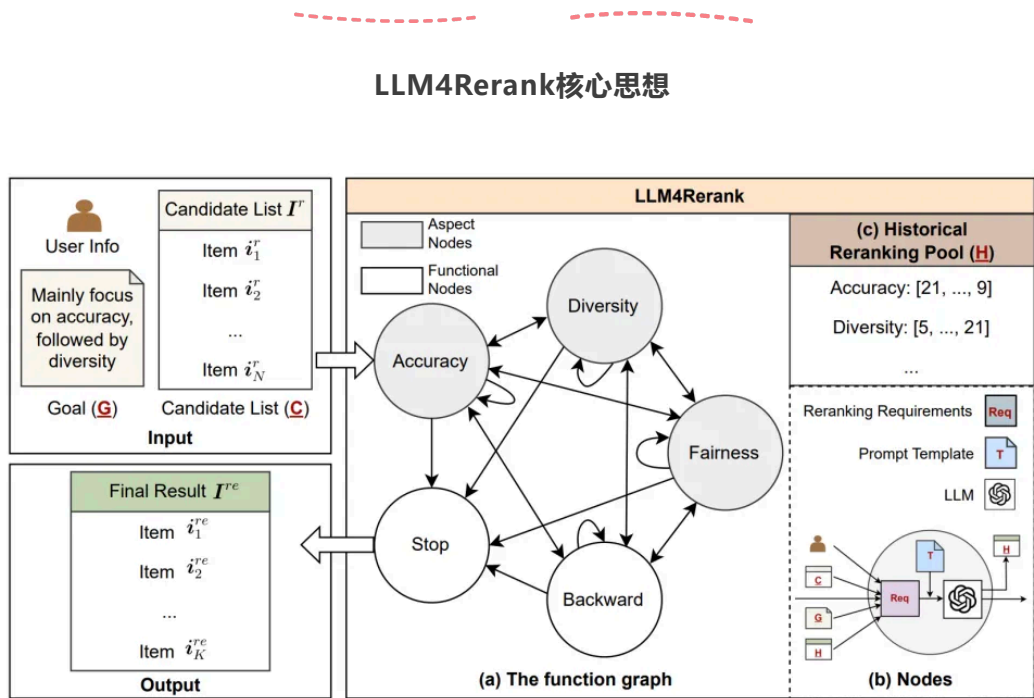


Figure 2: The overall structure of LLM4Rerank. Inputs are first directed to the “Accuracy” node, which initiates an automatic node-to-node reranking process in the function graph (a). Nodes (b) of varying colors represent the distinct aspects or functional steps within the reranking process, guiding the Large Language Model (LLM) in its deliberations. Once the “Stop” node is selected as the next node, a complete reranking process is considered finished. The final result is generated based on the most recent reranking result from the historical reranking pool (c). For simplicity, item lists in the historical reranking pool are represented by lists of item IDs. Note that the underlined symbols correspond to modules in subfigure (b)

1. 节点构建与功能设计

(1) 通用节点结构：

引入通用节点结构，每个节点代表在 LLM 考虑下的单步重排序操作，输入包含用户信息、候选项目、“Goal” 句子（定义重排序目标）和历史重排序池（记录过往节点结果），输出为当前节点的即时重排序结果及下一步节点指示，确保 LLM 能自动选择后续步骤，实现节点间的自动过渡，同时增强框架可扩展性，以便灵活应对不同方面需求。

(2) 方面节点设计：

设立 “Accuracy” “Diversity” “Fairness” 等方面节点，分别针对最终推荐列表的不同关键方面进行重排序任务。例如，“Accuracy” 节点着重强调用户与项目间的相关性，作为重排序起始点以确保基础准确性；“Diversity” 节点通过评估项目特定属性的变化程度（如使用 α -

NDCG 度量) 来提升推荐列表多样性; “Fairness” 节点将推荐结果的公平性操作化为不同样本组间的平均分数差异 (用 MAD 度量), 通过为推荐列表项目分配分数来计算 MAD 以评估公平性。

(3) 功能节点增强逻辑能力:

开发 “Backward” 和 “Stop” 两个功能节点, “Backward” 节点使 LLM 能在评估过程中选择性忽略不理想的重排序结果, 通过删除历史重排序池中的最新结果并按 LLM 输出指令前进到后续节点, 增强框架的逻辑调整能力; “Stop” 节点用于终止重排序输出序列, 当 LLM 指定此节点为下一步时, 提取历史重排序池中的最新结果作为最终重排序结果。

2. 自动重排序过程实现

Algorithm 1: The whole automatic reranking process of LLM4Rerank

Input: User information u , Candidate item list I^r , the reranking focus $Goal$, Maximum node count MC

Output: Final reranking result I^{re}

Note: $Function(a)(b)$ represents the execution of functions in node a with input b .

```

1: Initialize current node name  $CN = Accuracy$ ; Current
   reranking result  $CR = None$ ; Node count  $NC = 0$ ;
   Historical reranking pool  $Pool = []$ .
2: while  $CN \neq Stop$  do
3:    $CN, CR = Function(CN)(u, I^r, Goal, Pool)$ 
4:    $Pool.append(CR)$ 
5:    $NC++ = 1$ 
6:   if  $NC \geq MC$  then
7:      $CN = Stop$ 
8:   end if
9: end while
10: return  $Pool[-1]$ 

```

(1) 设置 “Goal” 引导个性化重排序:

引入手动输入的 “Goal” 句子作为重排序过程的初步输入之一, 明确特定重排序任务的主要关注点, 通过解读 “Goal” 与各节点的语义联系, 使 LLM 能自动选择最合适的节点, 满足个性化需求, 同时提升框架在不同场景下的可扩展性。

(2) 节点间自动过渡与条件控制:

每个节点在收到 LLM 回复后, 获取当前重排序结果及下一步节点指示, 实现节点间的流畅自动过渡。为避免长时间无响应或错误, 设定两个终止条件, 一是 LLM 自主选择 “Stop” 节点作为下一步, 二是达到预设的最大节点数 (排除 “Backward” 节点), 满足任一条件则完成重排序过程并输出最终结果。

3. 整体框架优势与效果验证

(1) **全面性能提升:** 通过上述设计, LLM4Rerank 框架能够在重排序过程中综合考虑准确性、多样性和公平性等多方面要求, 实验证明其在多个工业数据集上表现优异, 显著超越现有基线模型, 有效平衡了不同方面的需求, 提升了推荐系统的整体性能。

(2) **灵活个性化适应:** 借助 “Goal” 设置和自动重排序机制, LLM4Rerank 能够根据不同用户偏好和多样化的应用场景, 自动调整重排序策略, 实现个性化的推荐结果, 为用户提供更贴合需求的推荐服务, 增强用户体验。

节点prompt示例

1. “Accuracy” 节点

An Example of Accuracy Node Template

Considering a user: {User info}
Here's a list of the candidate movies: {Candidate List}
Your reranking goal: {Goal}
Your historical reranking: {Historical Reranking Pool}
Now, you need to focus on the accuracy aspect (the match between the user and items) and rerank the candidates based on the given information, and then give suggestions about the next step of reranking from the following reranking nodes considering the goal: {Available Nodes}
For your response format: {Format Description}

2. “Diversity”节点

An Example of Diversity Node Template

Considering a user: {User info}
Here's a list of the candidate movies: {Candidate List}
Your reranking goal: {Goal}
Your historical reranking: {Historical Reranking Pool}
Now, you need to focus on the diversity aspect (more items with different xx features should exist at the top of the reranking list) and rerank the candidates based on the given information, and then give suggestions about the next step of reranking from the following reranking nodes considering the goal: {Available Nodes}
For your response format: {Format Description}

3. “Fairness”节点

An Example of Fairness Node Template

Considering a user: {User info}

Here's a list of the candidate movies: {Candidate List}

Your reranking goal: {Goal}

Your historical reranking: {Historical Reranking Pool}

Now, you need to focus on the fairness aspect (For items with xxx feature value and items with xxx feature value, You should keep the average ranking of the two categories in the candidates similar) and rerank the candidates based on the given information, and then give suggestions about the next step of reranking from the following reranking nodes considering the goal: {Available Nodes}

For your response format: {Format Description}

4. “Backward”节点

An Example of Backward Node Template

Considering a user: {User info}

Here's a list of the candidate movies: {Candidate List}

Your reranking goal: {Goal}

Your historical reranking: {Historical Reranking Pool}

Now, you need to give suggestions about the next step of reranking from the following reranking nodes considering the goal: {Available Nodes}

For your response format: {Format Description}

END

推荐系统 4 语言模型 5 LLM与推荐 15 LLM论文阅读 13 #LLM学习 12

推荐系统 · 目录

上一篇

【LLM论文阅读】DMPO 与 S-DPO: LLM推荐排序模型的直接偏好优化研究

下一篇

【LLM论文阅读】MoRE: LLM通过多视角反思与迭代增强序列推荐