从头部到尾部：大语言模型增强的序列推荐与权衡优化

Adeep Hande[1], Shaghayegh Agah[1], Neeraj Sharma[1], Karina Halevy[2], Sardar Hamidian[1]1应用人工智能研究，Comcast 2卡内基梅隆大学计算机科学系 \{adeep_hande2,shaghayegh_agah,neeraj_sharma,sardar_hamidian\}@comcast.com,khalevy@cmu.edu

arXiv:2611.31052v1 [cs.IR] 23 Nov 2026

摘要

推荐系统存在流行度偏差，常常无法充分服务交互数据稀疏的长尾物品和冷启动物品。我们提出一种组特定数据增强方法，利用大语言模型生成合成交互，并将其表述为一个受约束的优化问题：在将整体 NDCG@20 的下降幅度限制在可接受阈值内的同时，最大化组特定 NDCG@20 和覆盖率。我们根据交互频率对项目进行划分，为每个组寻找最优增强策略。我们还研究了如何同时组合各组的增强，并发现，尽管各自的最优策略能够应对多组增强设计，但它们并不能简单相加。我们进一步对基于大语言模型的增强本身进行消融研究，在令牌约束下比较不同大语言模型规模和增强配置，发现增强质量会随上下文丰富度提升而提高，但成本也会按比例增加。在工业数据集上的 SASRec 实验表明，所提出的方法效果显著：极端长尾物品和普通长尾物品的组特定 NDCG@20 分别提升超过 100% 和 60%；在 MovieLens-32M 上则分别提升 40% 和 25%，同时提高了这些物品的覆盖率，且整体性能没有显著下降。

CCS 概念

• **计算方法** → 自然语言生成。

关键词

大语言模型、数据增强、流行度偏差、推荐

ACM 参考格式：

Adeep Hande[1], Shaghayegh Agah[1], Neeraj Sharma[1], Karina Halevy[2], Sardar Hamidian[1]. 2026年. 从头部到尾部：大语言模型增强的序列推荐与权衡优化。https://doi.org/XXXXXXX.XXXXXXX

1 引言

协同过滤模型从用户-项目交互中学习潜在项目表示，但当项目本身的交互数据稀疏时，模型表现不佳 [10, 27]。流行度偏差会形成反馈回路，系统性地抑制长尾和冷启动内容的曝光，因为这些内容缺乏足够的共现信号 [19, 37]。在我们的工业数据集（PRD）和 ML-32M [23], 中，排名前 15% 的项目分别占据全部交互的 95% 和 98%。基线 SASRec [13] 从头部到尾部的表现下降了 92%–95%（图 1），而冷启动物品则完全没有曝光。

传统方法（如自监督学习 [36],对比序列增强 [30], 和自适应负采样 [4] ）通过在现有交互分布内运作来应对交互稀疏问题，却没有为代表性不足的项目引入新的曝光。近年来，基于大语言模型的方法已被引入推荐范式中，用于生成成对序列 [28], 增强交互图 [31, 33], 基于元数据的物品增强 [3], 以及将大语言模型用作冷启动物品的推荐器 [22]。Kim等人 [14] 将大语言模型的词元空间与项目嵌入对齐；Huang等人 [11]基于筛选后的候选集模拟用户物品交互；Wang等人 [26]则对大语言模型推荐进行聚类。其他优化方法包括训练强化学习策略，以便在大语言模型增强中高效进行用户选择 [24]，以及对大语言模型进行后训练以实现冷启动推理[15]。上下文学习也被用于构建面向冷启动用户的合成交互矩阵 [2, 17]。

近期工作更多聚焦于解决长尾推荐问题，通过利用语义嵌入进行自蒸馏来改善协同信号 [18], ，以及采用检索增强型大语言模型 [29]。二者都在增强现有表示。为生成合成交互，TADA [5] 引入了具有尾部感知能力的增强方法，并采用启发式序列更新。由大语言模型生成的合成交互已被证明能够在大规模场景下作为可靠的增强信号 [8, 22, 26], ，这进一步表明大语言模型具备充分的推理能力，可以充当有效的数据增强器。然而，这些方法要么将长尾视为一个整体，要么未刻画增强条件下整体排名质量与群组特定排名质量之间的权衡。Wang等人 [25] 强调了探索冷启动物品的重要性，但未区分不同的流行度层级。基于此，我们针对交互分布引入基于百分位数的分组策略，利用从群组特定候选池中抽取的大语言模型生成交互来增强用户历史记录，并分析由此产生的

总体排序质量与分组特定排序质量之间的权衡。

我们的贡献总结如下：

- 我们提出了一种分组定向增强策略，根据交互频率对项目进行分类，为每个组量身定制增强方法，从而适应数据集的流行度分布。
- 我们通过估计帕累托前沿并考察总体指标与分组特定指标之间的相关区域，刻画二者之间的权衡关系，证明每个组都会产生一条独特的权衡曲线，而该曲线由其在历史数据中的代表性所决定。这表明，单一策略无法适用于所有组，而一刀切的方法并非最优。
- 我们证明，针对每个组分别达到最优的策略无法组合成全局最优解，从而确认跨组增强是一个需要联合校准的多目标优化问题。
- 我们针对候选选择、大语言模型规模、插入位置、增强比例以及

配置开展了全面的消融研究，提出了一种逐步实验框架，无需穷举搜索即可缩小设计空间。

2 方法论

2.1 数据集与分组策略

我们在 MovieLens-32M (ML-32M)[23], 和**专有真实世界数据集** (PRD) 上进行评估。PRD 包含来自大型娱乐平台的用户观看历史（电视节目、电影、体育等）。我们根据每个数据集的交互分布，在对数空间中计算其第 75th 和第 95th 百分位数，并据此将各数据集 I 中的物品目录划分为不同的物品交互频率组。我们根据基线SASRec在候选划分上的表现选择这些边界，因为这种划分产生的组在 NDCG@20 上存在显著差异：G_3 的表现更接近头部，G_2 的表现下降了近 95%，而 G_0/G_1 的得分则接近于零（图 1）。

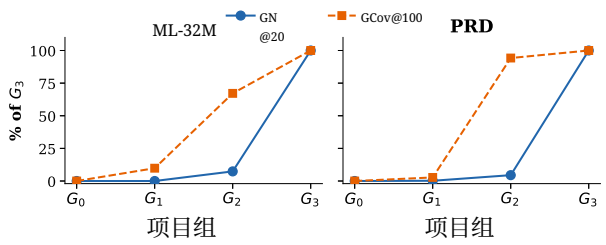

图 1：ML-32M（左）和 PRD（右）上各物品受欢迎程度组的原始组内 NDCG@20 (GN@20)

与通过任意交互阈值定义分组的固定阈值方法不同，这些基于百分位数的分组会适应各数据集的交互分布；相同的划分在 ML-32M 和 PRD 上对应不同的绝对阈值，反映出两者不同的受欢迎程度曲线。其中，G_1 (0-75th) 和 G_2 (75th-95th) 分别代表极端长尾物品和普通长尾物品，而 G_3 是热门层级，G_0 则是冷启动物品。

表 1：各数据集的项目组分布（完整标注数据集）及训练数据交互范围

组	百分位数	ML-32M		PRD	
		项目	范围	项目	范围
G_0	N/A	71	0	2,755	0
G_1	0-75th	62,645	[1-23]	225,806	[1-30]
G_2	75th-95th	16,414	[23- 731]	60,930	[30-302]
G_3	95th+	5,124	731+	16,671	302+

2.2 增强流程

Llama-3.3-70B-Instruct [9]用作默认的合成交互生成器。对于每个目标群体 $G_k \in \{G_0, G_1, G_2\}$，流程会随机抽取一定比例（$p$）的用户，并通过大语言模型为每位用户增强一个合成项目（$i = 1$），该项目从 G_k中抽取的 $c = 10$ 个候选项目中选出。每位用户仅增强一次的做法源于先前文献 [24, 26],：每位用户仅添加一条合成交互，既能最大限度降低用户偏好分布发生偏移的风险，又能为基础推荐器提供有意义的信号。

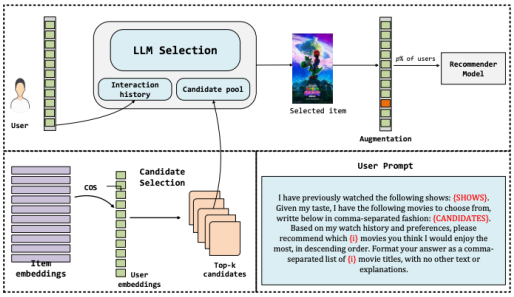

图 2：数据增强流程概览。

同时仍能为基础推荐器提供有意义的信号。候选选择依据 Linq-Embed-Mistral [12] 生成的项目嵌入与用户均值池化表示之间的余弦相似度进行；在 **PRD** 上采用哪种编码器，则由表2中的编码器消融实验决定。在 *ML-32M* 上，由于项目元数据不足以构建有意义的嵌入，因此通过随机抽样选择用于增强的候选项目。大语言模型从提示词（附录A）中列出的封闭候选池中进行选择，从而避免选出目录外的项目。选定的项目会插入用户交互序列 $S_u = (s_1, s_2, \ldots\ldots, s_n)$ 中位置 S_{n-2}，紧接在验证集（s_{n-1}）和测试集（s_n）项目之前。SAS-Rec 在增强后的交互矩阵上从头开始重新训练。所有指标均为三次独立运行的平均值。提示模板和实现细节见附录A。我们将评估表述为一个约束优化问题：最大化特定群体的 NDCG@20（GN@20）和群体覆盖率 GCov@100（衡量每位用户的前100条推荐中各群体所占的份额），同时要求整体 NDCG@20（N@20）的下降幅度维持在可接受的阈值内。

2.3 实验设计

鉴于搜索空间广阔，同时优化所有变量会导致组合爆炸。因此，按照先前的研究 [17, 24, 26]，我们固定了增强数据的数量和位置等特定变量，并优先报告影响最大的变化维度。为系统地探索这一搜索空间，我们采用坐标下降优化策略：依次固定超参数，以逐步分离并优化各个变量。这一贪心优化流程确保了在任何数据集和模型上的实际可复现性。下文将讨论旨在分离候选策略、增强比例和配置影响的序列化研究。**1）候选选择优化**：我们比较了 MTEB排行榜 [6, 20]中的嵌入编码器：Linq-Embed-Mistral（Mistral-7B） [12]，、qwen3-4b [32], 和 gte-large-en [16, 34]，并在 G_2 $p = 5\%$下选择使组内 NDCG@20 最大化的编码器。我们在多种增强比例下，将表现最佳的编码器与随机采样进行比较，并通过比较有无大语言模型重排序的配对设置，量化大语言模型的边际贡献（表2(b)）。对于 ML-32M，由于缺乏足够的项目元数据，我们随机采样候选项。

2) 通过增强比例进行帕累托前沿分析：给定**(1)**中最佳的候选选择策略，我们改变 p 以识别

各群体特定的帕累托前沿，找到权衡边界，以及目标指标和护栏指标同时改善的相关区域。

3) 交叉增强优化：根据行业目标，可能需要优化所有群组的性能，而不是只关注某个特定群组。受前一步的启发，我们在不同的增强划分下，同时增强 G_0、 G_1和 G_2，以评估最佳的跨组增强策略。

4) 大语言模型设计空间优化：我们评估在成本约束下，能否利用大语言模型优化技术进一步改进针对性解决方案。我们将 G_2 固定为 $p = 5\%$，并围绕大语言模型规模（8B、70B、405B）、插入位置（$S_{n/2}$ 与 S_{n-2}）、用户画像以及以大语言模型作为评判者 [7] 开展消融研究，以分离各组件对整体指标和分组指标的单独贡献。我们指出，此特定设置是一个示例性基准，用于展示优化框架；同样的方法也可应用于其他群组，或扩展至所有群组。

3 结果

1) 候选选择优化：表2（左）比较了在 PRD 数据集的 $p = 5\%$ 处，采用不同基于嵌入的模型对 G_2 进行增强的效果。编码器决定了提供给大语言模型的候选池质量，为其提供有依据的选择空间，并针对每位用户呈现最相关的项目。Mistral-7B 在整体性能与组别特定性能之间取得了最佳平衡，并且在所有 p 值下都持续优于随机选择（表2（右）），证实基于嵌入的选择能提升各项指标。与 Llama-70B 配合使用（表3），GN@20 进一步提升了三倍。随机选择会使两项指标均下降，因为从更大的词汇表中抽样，无法获得与用户语义相关的内容。*Mistral-7B*与 Llama-70B 的组合被用作所有下游实验的默认增强配置。仅采用基于嵌入的选择即可达到 +5.59% N@20 和 +15.97% GN@20（表2），这验证了余弦相似度已能筛选出相关候选项。在流程中加入大语言模型后，GN@20 提升至 +48.28%。此外，表3的前两行显示，大语言模型增强器在选择增强数据方面优于随机方法。

表2：候选选择优化（PRD）（G_2）：（左）仅编码器（右）$\Delta GN@20(\%)$：最佳编码器与随机。

嵌入	N@20	GN@20	p	随机	Mistral-7B
Mistral-7B	+6.36	+15.97	1%	-0.79	+ 8.14
Qwen-3B	+4.79	+6.42	5%	+24.87	+48.28
GTE-en	-6.47	+27.49	10%	+31.15	+62.42

2)通过增强比例进行帕累托前沿分析：如图3-a所示，每个组都呈现出一条独特的权衡曲线和最优点，这些曲线和最优点同时受部署目标指标和可接受的整体性能下降程度所决定。我们将 ≤ 5%的*N@20* 性能下降定义为用于识别可行配置的护栏约束。在PRD中， G_2 和 G_1 均呈现出正相关区域和权衡区域。在 G_2中，当增强比例为 $p = 1\%$时，整体

表3：在不同候选选择策略下，对比两种基线（无增强器）与大语言模型重排序 G_2： $p = 5\%$。

候选选择	增强器	ΔN@20(%)	ΔGN@20(%)
随机	-	-3.44	-13.04
随机	Llama-70B	-4.03	+24.87
mistral-7B	-	+5.59	+15.97
mistral-7B	Llama-70B	-2.09	+48.28

性能和组归一化折损累计增益指标分别提升了 +3.8%和 +8.14%，目标指标仍持续上升（$p = 15\%$中的组内新颖度（GN）为 +62%，组覆盖率（GCov）为 +5%），而N@20始终保持在基线的 −2%以内，这表明 G_2 是对增强最具容忍度的组。对已经具备部分学习表示的项目进行增强，可为SASRec提供更多信息性信号，使模型能够更好地学习其表示，同时不会引入足以降低整体性能的分布偏移。 G_1 项目在训练数据中极为稀疏，导致项目嵌入训练不足，容易受到少量合成增强的影响。在较低的 p下，合成交互可能不足以克服噪声，其影响会变得不稳定且更小。在更高的增强量（$p \geq 10\%$）下， G_1 呈现出持续的组级改进，但代价是牺牲整体性能。

ML-32M 在 G_1 和 G_2 图中未能提升 N@20，这可归因于候选选择策略。对于 G_1，由于测试用户数量有限（660），ML-32M 使用 GN@100 进行评估，因为在所有配置下，GN@20 的结果都接近于零。随机候选选择可能引入噪声，从而妨碍联合指标优化。由于不同数据集中的长尾组分布各异，目标指标的提升幅度可能有所不同；不过，各数据集上都呈现出一致的趋势。例如，在给定固定 p的情况下，两个数据集中，G_1 的指标提升都大于 G2，因为其基线显著更低。我们选择 $p = 3\%$作为 $G_1's$ 的运行点，以开展后续的跨组增强实验；该点位于相关区域内，且增强预算最低，有助于将总体增强比例降至最低 [5]。

表 4：基于 p 分布的 PRD 跨组增强最优点。

$[G_0(p)、\ G_1(p)、G_2(p)]$	N@20	G0=N@20	G1-N@20	G2-N@20
[5, 3, 15]	-24.49%	$2.96x$	+53.95%	**+45.67%**
[5, 5, 5]	-21.38%	$5.74x$	+11.31%	+19.86%
[1.66, 1.66, 1.66]	**-1.06%**	$2.87x$	-1.46%	+20.59%
[1.92, 1.16, 1.92]	-11.04%	$2x$	-7.69%	+38.41%
[3.84, 2.32, 3.84]	-14.95%	$4.17x$	-41.06%	+17.95%
[5, 3, 5]	-13.8%	$5.57x$	+32.24%	+19.24%
[5.77, 3.46, 5.77]	-20.6%	$6.61x$	**+71.52%**	13.44%

冷启动物品在整个训练过程中都保持随机初始化，不会从真实交互中获得梯度更新，而增强是这些嵌入唯一的学习信号来源。当增强幅度为 $p \leq 5\%$ 时，模型可以同时提升两项目标指标，并将 N@20 保持在可接受的阈值范围内。但超过这一幅度后，分布偏移会使权衡急剧加剧。最优

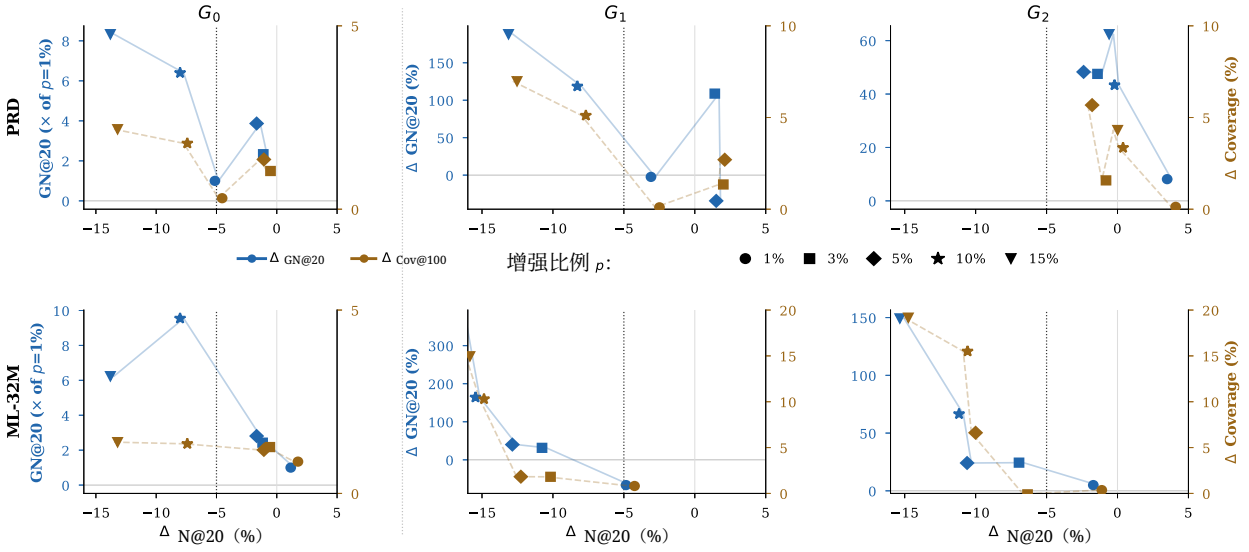

图3：不同（p）下 N@20-GN@20 和 N@20-GCov@100 的权衡关系。黑色点线表示护栏指标的预定义约束。以 G_k 的零基线为基准，相对于 1%（p）报告改进幅度。所有报告的数值均为三次运行的平均值，并具有统计显著性。

表5：PRD增强器消融性能（%）

增强器	ΔN@20（%）	ΔGN@20（%）	词元 (x)
Llama-70B-default（基线）	-2.09	+48.28	1.0x
Llama-8B	-0.15	+43.45	1.0x
Llama-405B	+3.11	+48.55	1.0x
Llama-70B（早期：$S_{n/2}$）	+0.06	+31.67	1.0x
以大语言模型作为评判者 [7]	**+6.36**	+34.93	2.06x
Llama-70B +用户画像	-0.16	+34.09	1.57x
Llama-70B + 元数据	+3.78	**+60.39**	5.12x

增强幅度为 $p = 5\%$，在提升 $4x$ GN@20 和 $2x$ GCov@100 的同时，导致 N@20 下降 $\leq 5\%$。在 ML-32M 上，由于增强的候选池小得多（71 对比 2.7k），只有在 $p = 1\%$ 时才能在 G_0 观察到相关区域。

3）交叉增强优化：我们评估各组增强策略能否以加法方式组合。直接组合各自独立的最优比例（总计 23%）会导致 N@20 下降 -24.49%。这一降幅可能源于增强总量本身、各策略间的交互作用，或两者兼有。我们将整体增强比例限制为 $\leq 15\%$，因为这代表我们在分组分析中评估的最大值（图 3）；这样可以将护栏指标 N@20 的下降控制在特定阈值以内，并衡量联合优化中的加法效应。

各组之间的非均匀 p 分配及其在增强总预算中的相应占比，是由单组权衡优化中观察到的模式所驱动的。在总交叉增强比例为 5% 且均等分配时，G_0 和 G_2 在第 3 行和第 4 行均有所提升，而非均匀分配带来了更大的总体增益。然而，G1组仍对分配比例较为敏感，在较高的 p 下会有所改善；这与我们的发现一致，即 G_1 需要达到最低增强量，才能产生稳定的增益。

没有任何配置能够在不导致整体性能下降的情况下同时改善所有组。基于前述结果应用按组的 p 后，倒数两行能够同时改善所有组的特定指标。然而，为了减轻护栏指标的下降，还需要进一步校准联合增强，将其定义为一个多目标优化问题。

4）大语言模型设计空间优化 为保持一致性，我们在所有消融实验中将 G_2 固定为5%的增强比例，并与表5所示的多种增强策略进行比较。插入位置对效果有显著影响：由于SASRec对较新的项目赋予较高权重，较早插入的增强内容会逐渐减弱其影响 [13, 21]。用户画像生成 [1] 和以大语言模型作为评判者 [7] 采用由简到繁提示 [35], 范式，利用中间结构迭代优化最终的候选选择，从而提升整体排序性能。在提示词中加入额外元数据可获得最高的GN@20，这表明更丰富的上下文信息有助于提高增强内容的质量。尽管这些增强配置带来了更大的提升，但最佳配置本身仍需在护栏指标与成本和计算可扩展性之间权衡。基础配置（Mistral-7B + Llama-70B）在成本与质量之间实现了最有利的平衡，适合生产部署；而元数据 enriquecement 则以显著更高的计算成本实现了最强的组特定提升。这促使我们对较小的大语言模型进行任务特定微调；与提示通用大语言模型相比，这种方法或许既能增强其推理能力、提升增强质量，又能降低推理成本[15]。

4 结论

我们提出了一种用于序列推荐的分组式大语言模型增强框架，并在ML-32M和一个大型专有数据集上进行了评估，将项目划分为冷启动（G0）、极端长尾（G1组）、普通长尾（G2组）和热门层级（G3）。围绕候选选择、增强比例、跨组划分和大语言模型端设计选择的实验结果表明，增强效果因组而异，并非由单一策略决定。在可以进行语义检索时，基于嵌入的候选选择比随机选择更可靠；代表性不足的各组在可接受的增强程度以及与整体NDCG@20之间的权衡方面也有所不同。大语言模型消融实验还表明，增强质量不仅取决于模型规模，也取决于提示设计和插入方式选择，这些因素会明显影响GN@20、覆盖率、成本和护栏性能。同样重要的是，单组设置下的最佳方案无法直接沿用于联合场景；在这种情况下，跨组增强仍是一个受约束的多目标问题。接下来，一个自然的方向是明确进行联合校准，并检验这些规律是否同样适用于其他推荐器和数据集。

参考文献

[1] Seunghwan Bang 和 Hwanjun Song。2025。基于大语言模型的推荐系统用户画像管理。*arXiv* 预印本 *arXiv:2502.14541*（2025）。[2] Shangkun Che、Minjia Mao 和 Hongyan Liu。2024。新型社区冷启动推荐：一种新颖的基于大语言模型的方法。（2024）。[3] Lei Chen、Chen Gao、Xiaoyi Du、Hengliang Luo、Depeng Jin、Yong Li 和 Meng Wang。2025。利用大语言模型增强基于ID的推荐。*ACM Trans. Inf. Syst.* 43，5，文章 112（2025年7月），30页。https://doi.org/10. 1145/3704263[4] Yongjun Chen、Jia Li、Zhiwei Liu、Nitish Shirish Keskar、Huan Wang、Julian McAuley 和 Caiming Xiong。2022。为序列推荐生成负样本。arXiv:2208.03645 [cs.IR] https://arxiv.org/abs/2208.03645[5] Yizhou Dang、Zhifu Wei、Minhan Huang、Lianbo Ma、Jianzhe Zhao、Guibing Guo 和 Xingwei Wang。2026。面向长尾序列推荐的长尾感知数据增强。载于*ACM Web*会议*2026*论文集。5886–5897。[6] Kenneth Enevoldsen、Isaac Chung、Imene Kerboua、Márton Kardos、Ashwin Mathur、David Stap、Jay Gala、Wissam Siblini、Dominik Krzemiński、Genta Indra Winata、Saba Sturua、Saiteja Utpala、Mathieu Ciancone、Marion Schaeffer、Gabriel Sequeira、Diganta Misra、Shreeya Dhakal、Jonathan Rystrøm、Roman Solomatin、Ömer Çağatan、Akash Kundu、Martin Bernstorff、Shitao Xiao、Akshita Sukhlecha、Bhavish Pahwa、Rafał Poświata、Kranthi Kiran GV、Shawon Ashraf、Daniel Auras、Björn Plüster、Jan Philipp Harries、Loïc Magne、Isabelle Mohr、Mariya Hendrik- sen、Dawei Zhu、Hippolyte Gisserot-Boukhlef、Tom Aarsen、Jan Kostkan、Konrad Wojtasik、Taemin Lee、Marek Šuppa、Crystina Zhang、Roberta Rocca、Mohammed Hamdy、Andrianos Michail、John Yang、Manuel Faysse、Aleksei Vatolin、Nandan Thakur、Manan Dey、Dipam Vasani、Pranjal Chitale、Simone Tedeschi、Nguyen Tai、Artem Snegirev、Michael Günther、Mengzhou Xia、Weijia Shi、Xing Han Lù、Jordan Clive、Gayatri Krishnakumar、Anna Maksimova、Silvan Wehrli、Maria Tikhonova、Henil Panchal、Aleksandr Abramov、Malte Ostendorff、Zheng Liu、Simon Clematide、Lester James Miranda、Alena Fenogenova、Guangyu Song、Ruqiya Bin Safi、Wen-Ding Li、Alessia Borghini、Federico Cassano、Hongjin Su、Jimmy Lin、Howard Yen、Lasse Hansen、Sara Hooker、Chenghao Xiao、Vaibhav Adlakha、Orion Weller、Siva Reddy 和 Niklas Muennighoff。2025。MMTEB：大规模多语言文本嵌入基准。*arXiv* 预印本 *arXiv:2502.13595*（2025）。https://doi.org/10.48550/arXiv.2502.13595[7] Francesco Fabbri、Gustavo Penha、Edoardo D'Amico、Alice Wang、Marco De Nadai、Jackie Doremus、Paul Gigioli、Andreas Damianou、Oskar Stål 和 Mounia Lalmas。2025。使用画像感知型大语言模型裁判评估播客推荐。载于第十九届*ACM*推荐系统大会论文集。1181–1186。[8] Natalija Glisovic、Martin Tegner 和 Danica Kragic。2026。利用大语言模型进行电子商务冷启动推荐的数据增强。载于欧洲信息检索会议。施普林格，94–99。[9] Aaron Grattafiori、Abhimanyu Dubey、Abhinav Jauhri、Abhinav Pandey、Ab- hishek Kadian、Ahmad Al-Dahle、Aiesha Letman、Akhil Mathur、Alan Schel- ten、Alex Vaughan 等。2024。Llama 3 模型家族。*arXiv* 预印本 *arXiv:2407.21783*（2024）。[10] Xiangnan He、Lizi Liao、Hanwang Zhang、Liqiang Nie、Xia Hu 和 Tat-Seng Chua。2017年。神经协同过滤。载于第 26 届万维网国际会议论文集（澳大利亚珀斯）（WWW '17）。国际万维网会议指导委员会，瑞士日内瓦共和国及州，CHE，173–182。https://doi.org/10.1145/3038912.3052569[11] Feiran Huang、Yuanchen Bei、Zhenghang Yang、Junyi Jiang、Hao Chen、Qijie Shen、Senzhang Wang、Fakhri Karray 和 Philip S Yu。2025年。用于冷启动推荐的大语言模型模拟器。载于第十八届 ACM 国际网络搜索与数据挖掘会议论文集（德国汉诺威）（WSDM '25）。美国计算机协会，美国纽约州纽约，261–270。https://doi.org/10 1145 3701551 3703546 . / . Jihoon Kwon、Junseong Kim、[12] Seolhwa Lee 等。2024年。Linq-Embed-Mistral：通过任务特定控制和质量优化，利用改进的 GPT 数据提升文本检索。Linq AI研究博客。https://getlinq.com/blog/linq-embed-mistral/[13] Wang-Cheng Kang 和 Julian McAuley。2018年。自注意力序列推荐。载于2018年 IEEE 国际数据挖掘会议（ICDM）。197–206。https://doi.org/10.1109/ICDM.2018.00035[14] Sein Kim、Hongseok Kang、Seungyoon Choi、Donghyun Kim、Minchul Yang 和 Chanyoung Park。2024年。大语言模型与协同过滤相遇：高效、全能的基于大语言模型的推荐系统。载于第 30 届 ACM SIGKDD 知识发现与数据挖掘会议论文集（西班牙巴塞罗那）（KDD '24）。美国计算机协会，美国纽约州纽约，1395 1406 10 1145 3637528 3671931 –。https://doi.org/ . / . Shijun Li、[15] Yu Wang 、 Jin Wang 、 Ying Li 、 Joydeep Ghosh 和 Anne Cocos。2025年。用于冷启动物品推荐的大语言模型推理。arXiv 预印本 arXiv:2511.18261（2025年）。[16] Zehan Li、Xin Zhang、Yanzhao Zhang、Dingkun Long、Pengjun Xie 和 Meishan Zhang。2023年。通过多阶段对比学习实现通用文本嵌入。arXiv 预印本 arXiv:2308.03281（2023年）。[17] Xurong Liang、Vu Nguyen、Vuong Le、Paul Albert 和 Julien Monteil。2025年。通过上下文学习解决序列电影推荐器中的用户冷启动问题。载于第十九届 ACM 推荐系统大会论文集（RecSys '25）。美国计算机协会，美国纽约州纽约，979–982。https://doi.org/10.1145/3705328.3748109[18] Qidong Liu、Xian Wu、Yejing Wang、Zijian Zhang、Feng Tian、Yefeng Zheng 和 Xiangyu Zhao。2024年。LLM-ESR：利用大语言模型增强长尾序列推荐。载于第 38 届神经信息处理系统国际会议论文集（加拿大不列颠哥伦比亚省温哥华）（NIPS '24）。Curran Associates Inc.，美国纽约州雷德胡克，文章 839，共 27 页。[19] Masoud Mansoury、Himan Abdollahpouri、Mykola Pechenizkiy、Bamshad Mobasher 和 Robin Burke。2020年。推荐系统中的反馈回路与偏差放大。载于第 29 届 ACM 国际信息与知识管理会议论文集（线上活动，爱尔兰）（CIKM '20）。美国计算机协会，美国纽约州纽约，2145–2148。https://doi.org/10 1145 3340531 3412152 . / . Niklas Muennighoff、Nouamane Tazi、Loïc Magne 和 Nils Reimers。2022年。MTEB：大规模文本嵌入基准。arXiv 预印本 arXiv:2210.07316（2022年）。https://doi.org/10 48550 2210 07316 . /ARXIV. . Jeonglyul Oh 和 Sungzoon [20] Cho。2024. 衡量序列推荐系统中的新近性偏差。arXiv 预印本 arXiv:2409.09722（2024年）。[21] Scott Sanner、Krisztian Balog、Filip Radlinski、Ben Wedin 和 Lucas Dixon。2023年。对于基于语言和物品偏好的冷启动推荐，大语言模型具有竞争力。载于第 17 届 ACM 推荐系统大会论文集（新加坡，新加坡）（RecSys '23）。美国计算机协会，美国纽约州纽约，890–896。https://doi.org/10.1145/ 3604915.3608845[23] Mark D. Smucker 和 Houmaan Chamani。2025年。扩展 MovieLens-32M 以提供新的评估目标。载于第 48 届国际 ACM SIGIR 信息检索研究与开发会议论文集（意大利帕多瓦）（SIGIR '25）。美国计算机协会，美国纽约州纽约，3520–3529。https://doi.org/10.1145/3726302.3730328[24] Nachiket Subbaraman、Jaskinder Sarai、Aniruddh Nath、Lichan Hong、Lukasz Heldt、Li Wei 和 Zhe Zhao。2025年。选择用户历史记录以生成用于冷启动物品推荐的大语言模型用户。arXiv:2511.21989 [cs.IR] https: //arxiv.org/abs/2511.21989[25] Dong Wang、Junyi Jiao、Arnab Bhadury、Yaping Zhang、Mingyan Gao 和 Onkar Dalal。2025年。以物品为中心探索冷启动问题。载于第十九届 ACM 推荐系统大会论文集（RecSys '25）。美国计算机协会，美国纽约州纽约，987–990。https://doi.org/10.1145/3705328.3748113[26] Jianling Wang、Haokai Lu、Yifan Liu、He Ma、Yueqi Wang、Yang Gu、Shuzhou Zhang、Ningren Han、Shuchao Bi、Lexi Baugher、Ed H. Chi 和 Minmin Chen。2024年。大规模推荐系统中的大语言模型用户兴趣探索。载于第 18 届 ACM 推荐系统大会论文集（意大利巴里）（RecSys '24）。美国计算机协会，美国纽约州纽约，872–877。https://doi.org/10.1145/3640457.3688161

[27] Xiang Wang、Xiangnan He、Meng Wang、Fuli Feng 和 Tat-Seng Chua。2019年。神经图协同过滤。载于国际 *ACM SIGIR* 信息检索研究与开发会议论文集（法国巴黎）（*SIGIR'19*）。美国计算机协会，美国纽约州纽约，165–174。https://doi.org/10.1145/3331184.3331267[28] Wei Wei、Xubin Ren、Jiabin Tang、Qinyong Wang、Lixin Su、Suqi Cheng、Junfeng Wang、Dawei Yin 和 Chao Huang。2024年。LLMRec：用于推荐的大语言模型与图增强。载于第 *17* 届 *ACM* 国际网络搜索与数据挖掘会议论文集（墨西哥梅里达）（*WSDM '24*）。美国计算机协会，美国纽约州纽约，806–815。https: //doi.org/10.1145/3616855.3635853

[29] Junda Wu、Chun Cheng、-Tong Yu、Zhankui He、Jianing Wang、Yupeng Hou 和 Julian McAuley。2024年。CoRAL：协同检索增强大语言模型提升长尾推荐。在第*30*届 *ACM SIGKDD* 知识发现与数据挖掘会议论文集 （西班牙巴塞罗那）（*KDD '24*）。美国计算机协会，纽约，NY，美国，3391–3401。https://doi.org/10.1145/3637528.3671901

[30] Xu Xie、Fei Sun、Zhaoyang Liu、Shiwen Wu、Jinyang Gao、Jiandong Zhang、Bolin Ding 和 Bin Cui。2022年。面向序列推荐的对比学习。在 *2022*年 *IEEE* 第*38*届国际数据工程会议（*ICDE*）。1259–1273。https://doi.org/10.1109/ICDE53745.2022.00099

[31] Lanling Xu、Junjie Zhang、Bingqian Li、Jinpeng Wang、Sheng Chen、Wayne Xin Zhao 和 Ji-Rong Wen。2025年。《挖掘大语言模型作为推荐系统的潜力：综合框架与实证分析》。*ACM* 知识发现与数据挖掘汇刊 19(5)，第105号文章（2025年6月），共51页。https: //doi.org/10.1145/3726871

An Yang、Anfeng Li、Baosong Yang 等 . 2025. Qwen3 技术报告 . arXiv:2505.09388 [cs.CL] https://arxiv.org/abs/2505.09388

[33] Haobo Zhang、Qianman Zhu 和 Zhicheng Dou。2025。通过用户偏好检索增强基于大语言模型的推荐重排序。载于第*31*届国际计算语言学会议论文集，OwenRambow、Leo Wanner、Marianna Apidianaki、Hend Al-Khalifa、Barbara Di Eugenio 和 Steven Schockaert（编）。计算语言学协会，阿布扎比，阿联酋，658–671。https://aclanthology.org/2025.coling-main.45/

[34] Xin Zhang、Yanzhao Zhang、Dingkun Long、Wen Xie、Ziqi Dai、Jialong Tang、Huan Lin、Baosong Yang、Pengjun Xie、Fei Huang 等。2024年。mgte：用于多语言文本检索的通用长上下文文本表示与重排序模型。载于*2024*年自然语言处理实证方法会议：工业界分会论文集。1393–1412。

[35] Denny Zhou、Nathanael Schärli、Le Hou、Jason Wei、Nathan Scales、Xuezhi Wang、Dale Schuurmans、Claire Cui、Olivier Bousquet、Quoc Le 等。2022年。由简单到复杂的提示设计使大语言模型能够进行复杂推理。*arXiv* 预印本 *arXiv:2205.10625*（2022年）。

[36] 周坤、王辉、Wayne Xin Zhao、朱宇涛、王思睿、张福正、王中原和 Ji-Rong Wen。2020年。S3-Rec：通过互信息最大化进行序列推荐的自监督学习。载于第*29*届美国计算机协会国际信息与知识管理会议论文集（虚拟会议，爱尔兰）（*CIKM '20*）。美国计算机协会，纽约州纽约，美国，1893–1902。https://doi.org/10.1145/3340531. 3411954

[37] 朱子薇、何云、赵星、张寅、Jianling Wang 和 James Caverlee。2021。协同过滤中的流行度—机会偏差。载于第*14*届*ACM*网络搜索与数据挖掘国际会议论文集（线上会议，以色列）（*WSDM '21*）。美国计算机协会，纽约，纽约州，美国，85–93。https://doi.org/10.1145/3437963.3441820

A 附录

A.1 流水线配置

表6展示了除非另有说明，否则所有实验均采用的默认配置。所有组件均可互换，我们的方法可以替换为任何基础推荐器、嵌入编码器或指令遵循型大语言模型。

表6：默认流水线配置。所有组件均可互换。

组件	实现
基础推荐器	SASRec [13]
大语言模型增强器	Llama-3.3-70B-Instruct [9]
候选编码器	Linq-Embed-Mistral [12]
候选选择	余弦相似度（top-c）
每位用户的项目数	$i = 1$
候选池大小	$c = 10$
插入位置	s_{n-2}

A.2 基于百分位数的项目分组

算法1描述了分组策略。该流程不依赖任何特定数据集，无需手动选择阈值即可应用于任何交互数据集。不过，最佳百分位数边界可能因领域和推荐系统架构而异，我们建议根据基线组级性能验证划分结果，以确保各层级彼此区分（图1）。

算法1基于百分位数的项目分组

要求： 训练集 D_{train}，测试/验证项目 $\mathrm{I}_{test} \cup \mathrm{I}_{val}$
确保： 项目组 $\{G_0, G_1, G_2, G_3\}$
统计每个项目的交互次数： $n_i \leftarrow |\{(u,i) \in \mathrm{D}_{train}\}|$
计算对数计数： $\ell_i \leftarrow \log(n_i + 1)$
移除离群值：排除满足 $|\ell_i - \mu| > 3\sigma$ 的项目
计算 P_{75}， P_{95} 剩余的 ℓ_i
$G_1 \leftarrow \{i : \ell_i \le P_{75}\}$
$G_2 \leftarrow \{i : P_{75} < \ell_i \le P_{95}\}$
$G_3 \leftarrow \{i : \ell_i > P_{95}\}$
$G_0 \leftarrow \{i \in (\mathrm{I}_{test} \cup \mathrm{I}_{val}) : n_i = 0\}$

G_0 项目通过合成交互被引入训练历史，从而为模型提供这些项目最初的学习信号。评估衡量模型在留出的测试序列上的排序质量，并且在数据增强步骤中不会看到任何测试交互。

A.3 提示模板

以下列表式提示模板用于所有基于大语言模型的候选生成实验。用户交互历史（节目）和候选项目会根据用户分别填充。

```
我之前看过以下节目：{SHOWS}。根据我的喜好，以下是可供我选择的电影，按逗号分隔：{CANDIDATES}。根据我的观看历史和偏好，请按喜爱程度从高到低推荐你认为我最可能喜欢的{i}部电影。请将答案格式化为以逗号分隔的{i}个电影名称列表，不要添加其他文字或说明。
```

图4：用于基于大语言模型的候选项选择的列表式提示模板。

A.4 覆盖率分析

表7列出了根据分组分析得出的各组最佳增强比例（p^*）对分组覆盖率的影响。对选定帕累托最优点（图3-b）进行的 PRD 覆盖率分析显示，增强会将曝光从 G_3 重新分配至目标组，从而提高推荐结果中的物品多样性。在 PRD 中增强 G_1 以及在 ML-32M 中增强 G_0 也会提高 G_3 覆盖率；这是因为合成交互同时提升了总体指标和分组指标。在 $\leq 5\%$ 的 N@20阈值下，ML-32M 的 G_1 和 G_2使最优 p^* 仅限于 $p = 1\%$，导致覆盖率变化可以忽略不计。

表 7：每个群体中所选帕累托最优（p）对应的分组覆盖率分布偏移（G_k（$p*$））。每一行展示了增强目标群体时所有群体的覆盖率变化。

	$G_k\,(p^*)$	ΔG_0	ΔG_1	ΔG_2	ΔG_3
PRD	G_0 (5%)	$^{+1.35}$	-0.38	$^{+0.40}$	-1.38
	G_1 (3%)	0.0	$^{+1.35}$	-2.77	$^{+1.43}$
	G_2 (5%)	0.0	$^{+0.46}$	$^{+5.68}$	-6.14
ML	G_0 (3%)	$^{+1.26}$	-3.92	-2.78	$^{+5.43}$
	G_1 (1%)	0.0	$^{+0.81}$	-5.16	$^{+4.35}$
	G_2 (1%)	0.0	-0.90	$^{+0.36}$	$^{+0.54}$

A.5 数据预处理

对于两个数据集，我们均遵循 SASRec 风格的训练与评估方式 [13], ，倒数第二次交互用于验证，最后一次交互用于测试，其余交互用于训练。交互序列按时间顺序排列，并截断或补零至最大长度 150。交互次数少于 K 次的用户将从两个数据集中排除。