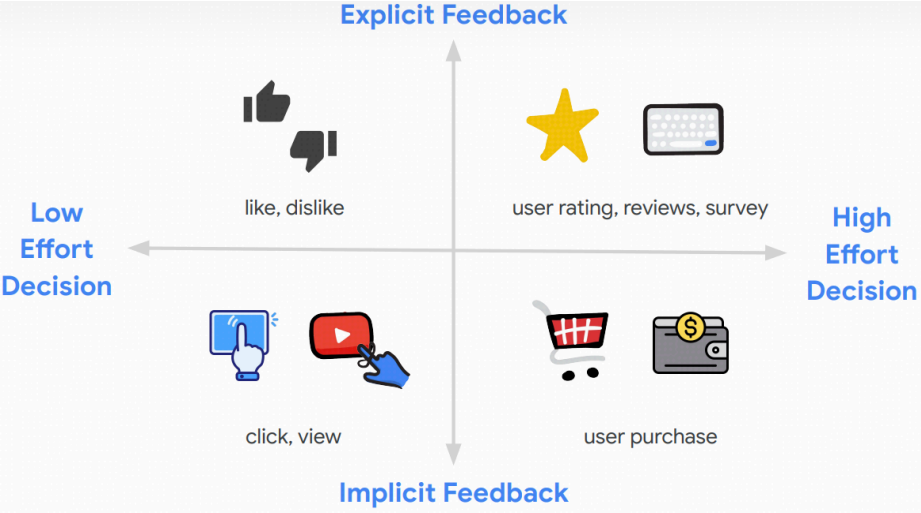


2024谷歌：运用LLM的COT链式思维策略，深度优化推荐系统效果

SmartMindAI
专注搜索、广告、推荐、大模型和人工智能最新技术，欢迎关注我
已关注

Introduction

大型语言模型（LLMs）的快速发展展示了它们在一系列应用中的潜力。特别是，零样本链式思维提示的出现为这些模型提供了进行多步推理的途径。相比之下，推荐系统（RecSys）的场景提出了一个微妙的挑战，其中推理不仅超越了客观标准，还涉及了主观性和个性化用户偏好的范畴。这一方面在利用 LLMs 的推理能力方面仍然是一个未充分探索的领域。 先前的研究已经在推荐系统中利用了 LLMs，采用了上下文学习和指令调整等技术。然而，如何全面理解 LLMs 在个性化偏好背景下执行推理的问题仍然不明确。

为了克服这个问题，我们提出了 Rec-SAVER 框架。这一框架提供了一种有效的方法来评估 LLMs 输出的质量，有助于我们理解 LLMs 在个性化推荐场景中的推理动态。我们通过评估输出的一致性、忠实度和洞察力来衡量人类的评估。我们的观察表明，诸如BLEU 和 ROUGE 这样的语法指标适合评估 LLMs 输出的忠实度。相反，METEOR 和 BERTScore 等指标在衡量生成输出的一致性方面更为有效。到目前为止，这是首次全面研究 LLMs 推理效果和质量对个性化 RecSys 任务的影响的尝试。

Methodology

$i \in \mathcal{I}$的整体满意度。每个物品i都关联有元数据$^+\mathcal{M}_i$，包含标题、描述、类别、品牌和价格等细节。用户的购买历史 $\mathcal{H}_u = (h_{u,1}, h_{u,2}, \ldots, h_{u,t})$ 构成了一组按时间顺序排列的过去购买记录。每个过去的购买$h_{u,j}$表示一个包含三个元素的三元组$^+$，包括购买的物品j的元数据、用户对该购买物品的评分，以及用户对该物品的评论。 评分预测任务的主要目标是预测用户尚未评论的物品的未知评分，从而填补这些物品的用户反馈空白。目标的形式化表述如下：

- 目标：预测用户尚未评论的物品的未知评分$\hat{\mathbf{r}}_{u,i}$，其中$\hat{\mathbf{r}}_{u,i}$是用户u对物品i的预测评分$\mathcal{R}$是用户评分的集合$\mathcal{D}$是用户评论的集合$\mathcal{M}_i$是物品i的元数据$\mathcal{H}_u$是用户u的购买历史。
- 目标公式： $\hat{\mathbf{r}}_{u,i} = f(\mathcal{M}_i, \mathcal{H}_u)$

其中f是一个函数，用于根据物品的元数据$\mathcal{M}_i$和用户的历史购买记录$\mathcal{H}_u$预测未知评分。

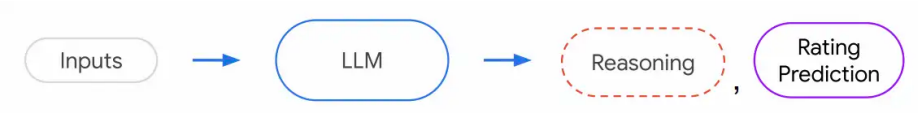
$$\hat{\mathbf{r}}_{u,i} = \arg\max_k \mathbb{P}(\mathbf{r}_{u,i} = k \mid \mathcal{H}_u, \mathcal{M}_i), \quad \text{where } i \notin \mathcal{H}_u,\ k \in \{1,2,3,4,5\}.$$

其中$\hat{\mathbf{r}}_{u,i}$表示用户u对内容i的预测评分，其中预测为1到5之间的评分。这个预测基于用户购买历史$\mathcal{H}_u$和内容的元数据$\mathcal{M}_i$，其中内容i是用户u的未评价内容。

推荐系统领域的最新进展利用 LLMs 来建模公式描述的评分预测任务，表示为：

$$\hat{\mathbf{r}}_{u,i} = \text{LLM}(\mathcal{H}_u, \mathcal{M}_i)$$

Zero-shot Learning with Reasoning



采用零样本CoT策略进行提示，我们利用语言模型来生成针对特定用户u和推荐内容i的推理响应$\hat{\mathbf{s}}_{u,i}$，同时预测一个评分$\hat{\mathbf{r}}_{u,i}$。如图所示，这一过程旨在通过语言模型的输出，为用户提供基于推荐内容的个性化推理和评估，从而提升推荐系统的交互性和用户满意度：

$$\langle \hat{\mathbf{s}}_{u,i},\ \hat{\mathbf{r}}_{u,i} \rangle = \text{LLM}(\mathcal{H}_u, \mathcal{M}_i).$$

我们的提示包含四个关键要素：一个前言，用户历史$\mathcal{H}_u$，新内容元数据$\mathcal{M}_i$，以及一个任务$^+$描述。前言提供了后续信息的上下文，并建立了从1到5的评分尺度。

Table 1: Abstract prompt template guiding our zero-shot approach, prompting the model to reason by leveraging user history and inferring preferences before making predictions.

Preamble	*e.g.* Here is information about a user and a new product ...
User History	$h_{u,1} = (\mathcal{M}_1, \mathbf{r}_{u,1}, \mathbf{d}_{u,1}), \ldots, h_{u,t} = (\mathcal{M}_t, \mathbf{r}_{u,t}, \mathbf{d}_{u,t})$
New Item	$\mathcal{M}_i$, *e.g.* title, brand, category, ...
Task Description	*e.g.* Given the user's past purchase history [...] how they will rate the new item? [...] After your reasoning, predict a numerical rating.

在前言之后，以序列方式呈现用户历史$\mathcal{H}_u$，详细描述用户的过去互动。随后，引入新内容i及其元数据信息$\mathcal{M}_i$，在任务描述之前，促使模型进行预测。任务描述还明确了模型响应的输出要求。前言的抽象模板在表中展示，而更多的提示细节则在附录中提供。与传统的RecSys建模技术不同，我们采用自然语言来呈现所有信息的方法，这使得丰富的内容以自然语言的形式得到更直观的表示，而不是仅用数字ID，从而能够更全面地理解信息。

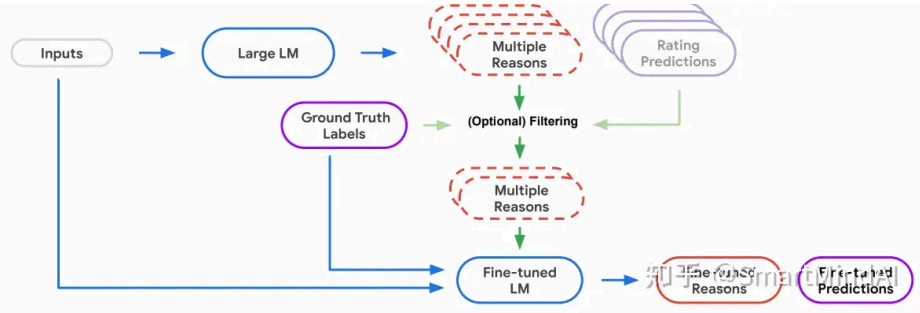
Fine-tuning with Reasoning

零样本学习结合思考过程提示可能会很耗费计算资源。因此，利用特定领域的数据集来微调模型成为了一种实用的方法，特别是在使用较小的预训练模型时。我们感兴趣的是研究通过推理输出进行

数 T > 0 来收集多个推理响应和评分预测：

$$\langle \hat{\mathbf{s}}_{u,i}^m,\ \hat{\mathbf{r}}_{u,i}^m \rangle = \mathrm{LLM}(\mathcal{H}_u, \mathcal{M}_i),$$

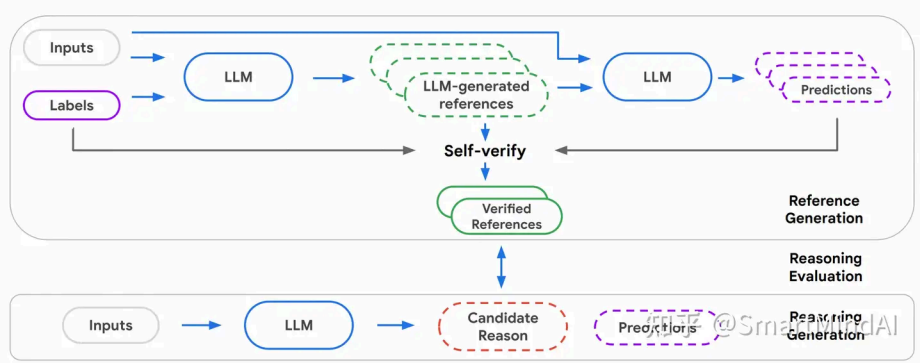
其中$m = 1, 2, \ldots, M$表示从解码器+中采样得到的M个候选输出对的索引。这个过程产生了一组多样的推理路径，这对于个性化推荐特别有利，认识到相同的评分可能源自不同的个人偏好和原因。可选地，我们可以使用不同的方法来筛选出与实际事实不一致的推理响应$\hat{\mathbf{r}}_{u,i}^m$整体方法的示意图+如图所示，其中使用推理响应 $\hat{\mathbf{s}}_{u,i}^m$ 和真实的实际评分标签$\mathbf{r}_{u,i}$作为目标进行训练。



Rec-SAVER: Evaluation of Reasoning

与解决数学问题或一般问答任务的推理过程形成鲜明对比，推荐系统+（RecSys）的评分预测推理高度主观且针对个体，旨在为单个用户设计。与在其他领域中人类可以提供推理步骤并验证其有效性，从而产生经过精心挑选的参考不同，在RecSys领域，这样的参考难以获得，由于用户偏好的主观性。为解决这一挑战，我们提出了一种名为**Rec-SAVER**的系统：**推荐**系统**自动**验证和**评价**推理。Rec-SAVER旨在自动生成适用于优质推理的参考，这些参考随后可以用于定量评估由LLMs生成的推理回答的质量。此外，我们进行了一项人类研究，以展示我们的方法与实际人类判断的匹配度，从而为Rec-SAVER在评估RecSys应用推理质量的有效性提供验证。

Reference Generation with Self-Verification



Rec-SAVER涉及两步流程，利用由LLM生成的解释和LLM的自我验证。如图所示，我们向LLM提供用户-内容对 $(\mathcal{H}_u, \mathcal{M}_i)$。同时，目标用户评分$\mathbf{r}_{u,i}$也被作为输入提供。模型被指示提供事后解释，描述为什么用户根据给定的用户历史和新内容信息，会给予这样的评分。我们将由LLM生成的这种事后解释表示为$\hat{\mathbf{g}}_{u,i}$。

类似于前面的方法，我们生成N个不同的解释 $\hat{\mathbf{g}}_{u,i}^n$，其中$n = 1, 2, \ldots, N$。然后，这些事后解释被传递到验证流程。

为了确保由LLM生成的解释 $\hat{\mathbf{g}}_{u,i}^n$ 的可信度和一致性，我们在之前的解释生成流程之上实施了一个自我验证步骤。自我验证步骤包括对相同的LLM进行第二次调用，输入用户-内容信息 $(\mathcal{H}_u, \mathcal{M}_i)$ 和之前调用生成的解释$\hat{\mathbf{g}}_{u,i}^n$。然后，模型被任务基于用户历史、新内容信息和事后解释，做出评分预测，形式上定义为

$$\tilde{\mathbf{r}}_{u,i}^n = \mathrm{LLM}(\mathcal{H}_u, \mathcal{M}_i, \hat{\mathbf{g}}_{u,i}^n)$$

知乎

为了避免$\hat{\mathbf{g}}_{u,i}^n$直接泄露真实评分，我们采用了一个简单的后处理步骤，在进行预测之前，删除提及"*评分为*", "*星级*", 或"*分数*"的句子。在未来的工作中，我们旨在改进这一手动流程，以完全确保信息泄露的消除。然后，我们验证新的评分 $\tilde{\mathbf{r}}_{u,i}^n$ 是否与原始ground truth$\mathbf{r}_{u,i}$相匹配。通过自验证步骤筛选出的解释$\hat{\mathbf{g}}_{u,i}^n$被保留作为最终验证的参考，形成了一池由LLM生成的多样化参考$\hat{\mathcal{G}}$。这个两步过程遵循了直觉，即基于给定信息和ground truth[+]的好解释应该使模型能够做出正确的预测。通过将基于生成解释的预测验证与ground truth评分进行对比，我们确保只有高质量的解释被保留作为最终参考。这些验证过的参考随后作为未知的黄金参考集$\mathcal{G}$的代理。由于自验证可能根据不同样本产生不同的最终参考，每个样本的最终参考数量可能会有所不同。

Algorithm 1 Reference generation with self-verification

Inputs: N
$\hat{\mathcal{G}} \leftarrow \emptyset$ ▷ verified references
for $(\mathcal{H}_u, M_i, \mathbf{r}_{u,i})$ in dataset **do**
  for $n = 1 \ldots N$ **do**
    $\hat{\mathbf{g}}_{u,i}^n \leftarrow \text{LLM}(\mathcal{H}_u, \mathcal{M}_i, \mathbf{r}_{u,i})$
    $\hat{\mathbf{g}}_{u,i}^n \leftarrow \text{post-process}(\hat{\mathbf{g}}_{u,i}^n)$
    $\tilde{\mathbf{r}}_{u,i}^n \leftarrow \text{LLM}(\mathcal{H}_u, \mathcal{M}_i, \hat{\mathbf{g}}_{u,i}^n)$
    if $\tilde{\mathbf{r}}_{u,i}^n = \mathbf{r}_{u,i}$ **then**
      $\hat{\mathcal{G}} \leftarrow \hat{\mathcal{G}} \cup \{\hat{\mathbf{g}}_{u,i}^n\}$
    end if
  end for
end for

Experiments

在亚马逊[+]产品评论数据集上进行实验，这个数据集提供了全面的用户反馈，包括评分和评论文本，以及产品元数据的详细信息，如描述、类别信息、价格和品牌等。我们专注于评分预测任务，在美妆和电影/电视这两个不同的领域中进行评估。

Zero-shot Learning Improves with Reasoning

Task Description

Given the user's past purchase history [...] how they will rate the new item?
[...] After your reasoning, predict a numerical rating.
=== Please follow the format below: ===
Reason
Write your reasoning explanation here.
Rating
Give a single numerical rating, e.g. 1

首先，我们探讨了在预测之前提示模型进行推理（零射链式思维）的效果与直接预测（零射）之间的差异。表中展示了零射链式思维输入提示的最终任务描述与直接预测的差异。

Beauty	Naive Baseline (Avg.)	0.52	0.60	0.25	1.35	1.75	-	-	-	-
	Our Method (zero-shot CoT)	**0.56**	**0.62**	**0.37**	**1.14**	**1.60**	0.236	0.503	0.339	0.665
	- No Reasoning Outputs	0.49	0.57	0.23	1.35	1.70	-	-	-	-
	- No Review	0.48	0.57	0.21	1.35	1.69	**0.237**	**0.507**	0.337	**0.667**
	- No Review, No Rating	0.43	0.53	0.19	1.42	1.75	0.215	0.494	0.331	0.660
	- No Item Description	0.48	0.57	0.21	1.33	1.66	0.235	0.504	**0.340**	**0.667**
	One-shot	0.43	0.57	0.26	1.52	1.97	0.225	0.502	0.335	0.664
Movies/TV	Naive Baseline (Avg.)	0.59	0.63	0.30	1.21	1.63	-	-	-	-
	Our Method (zero-shot CoT)	**0.62**	**0.66**	**0.40**	**1.06**	**1.53**	**0.194**	**0.465**	**0.296**	**0.647**
	- No Reasoning Output	0.59	0.63	0.29	1.18	1.56	-	-	-	-
	- No Review	0.58	0.63	0.28	1.20	1.58	0.173	0.452	0.291	0.641
	- No Review, No Rating	0.43	0.54	0.20	1.42	1.75	0.150	0.434	0.283	0.633
	- No Item Description	0.54	0.62	0.28	1.22	1.60	0.188	0.455	0.296	**0.647**
	One-shot	0.47	0.59	0.23	1.32	1.68	0.182	0.452	0.276	0.641

随后，表列出了不同零射消融实验的结果对比。我们也与一个朴素基准进行了比较，即使用用户历史评分的平均值作为对未来内容的预测。我们观察到了显著的性能提升，在两个产品领域中，当模型被引导在预测的同时输出推理时（Our Method (零射链式思维) vs.（无推理输出）。这表明，对于大规模语言模型来说，解决个性化任务在没有进一步指导的情况下（如进行中间推理步骤）是固有困难的。

Fine-tuning with Reasoning Data

我们利用Flan-T5模型进行微调实验。尽管所有这些模型都是在多种数据和任务上进行训练，但值得注意的是，PaLM-2-M在常见的基准测试[4]上报告的质量显著高于Flan-T5，包括大规模多任务语言理解（MMLU）。除非特别说明，我们采用Flan-T5 XL模型（参数量为3B）进行输出推理并预测最终评级。

	Model Size	Reas-oning	Binary Acc.	Binary F1	Binary AUC	Multi. Acc.	Multi. AUC	Multi. MAE ↓	Multi. RMSE ↓	BLEU	ROUGE-1 F1	METEOR	BERT Score
Beauty	Small	✓	0.62	0.53	0.65	0.30	0.63	1.35	1.84	0.225	0.499	0.342	0.663
	Base	✓	0.59	0.47	0.66	0.27	0.64	1.37	1.83	0.239	0.507	**0.344**	**0.667**
	Large	✓	0.64	0.59	0.67	**0.33**	0.65	1.26	1.73	0.240	0.506	0.343	0.666
	XL	✓	**0.67**	**0.61**	**0.78**	0.30	**0.69**	**1.24**	**1.68**	**0.241**	**0.510**	0.339	**0.667**
	XL	✗	0.55	**0.61**	0.74	0.28	0.67	1.31	1.75	-	-	-	-
	XL (no fine-tuning)	✗	0.55	0.40	0.56	0.22	0.56	1.64	2.09	-	-	-	-
Movies/TV	Small	✓	0.60	0.55	0.70	0.33	0.66	1.23	1.71	0.137	0.423	0.272	0.627
	Base	✓	**0.65**	0.59	**0.72**	**0.34**	**0.68**	1.18	1.65	0.153	0.438	0.279	0.634
	Large	✓	0.64	0.58	**0.72**	0.32	0.67	1.23	1.70	**0.165**	0.448	**0.286**	0.639
	XL	✓	**0.65**	**0.61**	**0.72**	**0.34**	0.67	**1.17**	**1.64**	**0.165**	**0.449**	**0.286**	**0.643**
	XL	✗	0.62	0.57	0.70	0.33	0.66	1.27	1.75	-	-	-	-
	XL (no fine-tuning)	✗	0.61	0.43	0.61	0.23	0.61	1.56	2.00	-	-	-	-

从两个领域的一致结果来看，微调模型在预测前进行推理，可以显著提高所有指标的性能。

Human Judgment Alignment Analysis

我们设计了一个人类判断对齐研究，以评估Rec-SAVER的有效性。我们评估了四种常用的自然语言生成（NLG）指标：BLEU、ROUGE-1、METEOR 和 BERTScore。BLEU 和 ROUGE-1 通过计算生成输出和参考文本之间的确切n-gram交集来衡量语法相似性。相比之下，METEOR 和 BERTScore 考虑了语义相似性，通过整合上下文信息提供更全面的评估。表显示一致性与所有NLG指标之间存在一致的正相关，表明我们提出的方法使用语言模型生成的参考文本与一致性很好地对齐。然而，新颖性与BLEU之间没有相关性，与ROUGE-1 F值之间的相关性较低，但与METEOR 和 BERTScore之间存在轻微的正相关。与"一致性"和"忠实度"不同，"新颖性"是一个探索性指标，旨在理解语言模型如何"让人类评估者感到惊喜"。

human annotated scores.

	Mean	Cohen κ	Avg. ρ	p-value
Coherence	3.72	0.37	0.37	1e-10
Faithfulness	0.63	0.63	0.63	1e-12
Insightfulness	2.80	0.33	0.34	6e-4

Table 7: Correlation between coherence, insightfulness, and automatic NLG metrics. The annotated scores are averaged across the annotators for each sample.

	Coherence	Insightfulness
BLEU	0.36	0.02
ROUGE-1 F1	0.40	0.10
METEOR	0.40	0.25
BERTScore	0.36	0.29

为了验证Rec-SAVER中的自我验证步骤的有效性，我们将带有和不带有参考自我验证步骤的计算指标进行了比较。表格数据显示，从自我验证参考中计算出的指标与coherence分数的相关性更强，这表明自我验证有助于提高LLM生成参考的可信赖度。综合这一部分的所有结果，我们发现我们的提议Rec-SAVER推理评价方法与人类对质量的评估之间呈现很强的一致性。

原文《Leveraging LLM Reasoning Enhances Personalized Recommender Systems》

发布于 2024-09-05 14:29 · IP 属地北京

推荐系统 LLM 谷歌 (Google)