

【LLM论文阅读】RankGPT：利用 GPT 模型进行排序任务及蒸馏

原创 方方 方方的算法花园 2024年10月11日 14:10 北京

论文标题：Is ChatGPT Good at Search? Investigating Large Language Models as Re-Ranking Agents
论文链接：<https://arxiv.org/pdf/2304.09542>
论文作者所在机构：山东大学、百度、莱顿大学
Github链接：<https://github.com/sunnweiwei/RankGPT>
一句话概括：利用GPT模型进行文本排序任务，将GPT模型的标注结果用于模型蒸馏

创新点

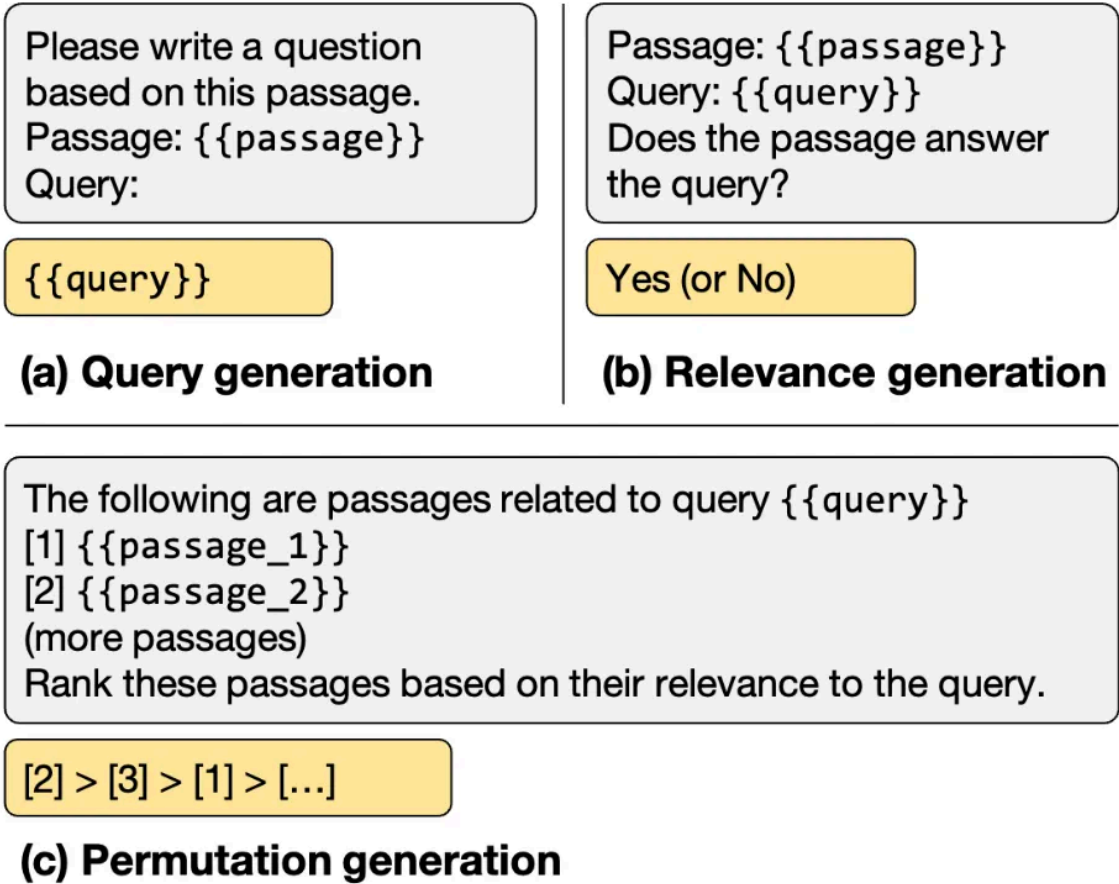
1. **对 LLMs 用于段落重排任务的指令方法进行了研究，并引入了一种新的排列生成方法**
 - a. 提出了一种指令排列生成方法，直接让 LLMs 输出一组段落的排列，克服了现有方法在重排时性能有限且依赖模型输出的对数概率的问题。
 - b. 采用了滑动窗口策略来解决 LLMs 的上下文长度限制问题。
2. **对 ChatGPT 和 GPT - 4 在各种段落重排基准上进行了综合评估，包括一个新提出的 NovelEval 测试集**
 - a. 使用了 TREC、BEIR 和 Mr.TyDi 三个成熟的 IR 基准数据集进行评估。
 - b. 收集了一个新的测试集 NovelEval，包含 21 个新问题，用于评估模型对未知知识的排名能力，以解决 LLMs 数据污染的担忧。
3. **提出了一种蒸馏方法，用于学习具有 ChatGPT 生成的排列的专用模型**
 - a. 介绍了一种排列蒸馏技术，将 ChatGPT 的段落排名能力模仿到一个更小的专用排名模型中。
 - b. 通过实验证明了蒸馏后的 440M 模型在 BEIR 基准上优于 3B 的监督模型，展示了该方法的成本效益优势。

主要方案

基于 LLMs 的段落重排方法

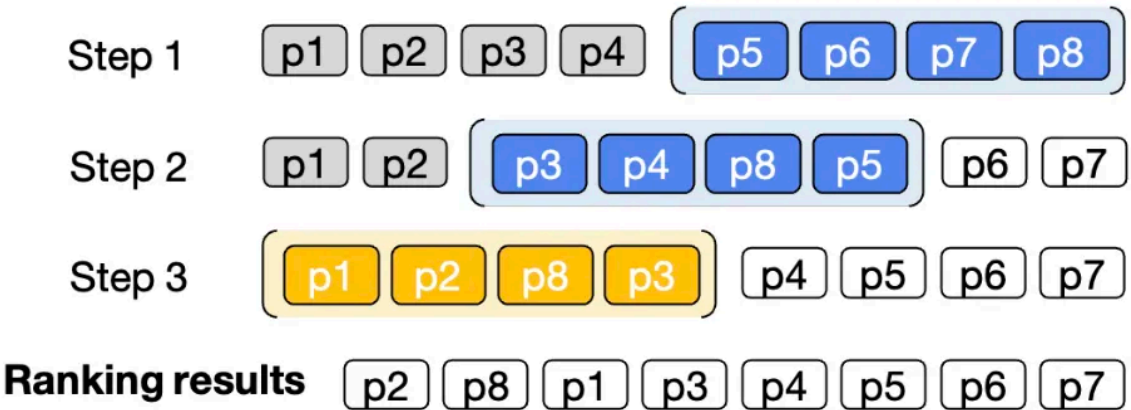
• 指令排列生成

将一组段落输入到 LLMs 中，要求 LLMs 根据段落与查询的相关性按降序生成段落的排列，直接对段落进行排名，不产生中间相关性分数。（下图中的c图）



• 滑动窗口策略

由于 LLMs 的token限制，只能对有限数量的段落进行排名。该策略通过设置窗口大小w和步长s，对段落进行重新排序。首先对 (M-w) 到M的段落进行排名，然后滑动窗口，对 (M-w-s) 到 (M-s) 的段落重新排名，重复此过程直到所有段落都被重新排名。



排列蒸馏技术

• 排列蒸馏方法

从 MS MARCO 数据集中随机采样 10,000 个查询，每个查询用 BM25 检索 20 个候选段落。将 ChatGPT 预测的排列作为目标，通过 RankNet - based 蒸馏目标将其蒸馏到一个学生模型中，减少学生模型和 ChatGPT 排列输出之间的差异。

• 训练目标

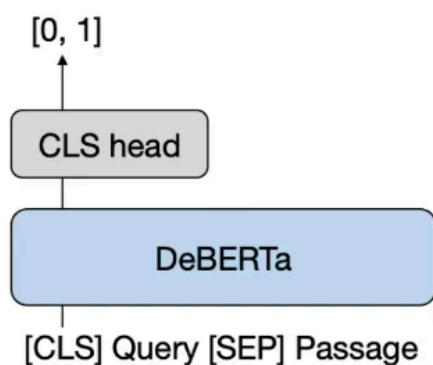
训练目标具体计算方式：假设有一个query q和M个候选段落(p1,...,pM)，利用chatgpt生成M个段落的排序结果，记为R = (r1,...,rM)，其中 ri ∈ [1,2,...,M]是段落 pi 的排名，ri=3 表

示 p_i 在 M 个段落中排名第三。此时有个目标训练模型 $s_i = f_{\theta}(q, p_i)$ ，使用交叉编码器来计算配对 (q, p_i) 的相关度得分 s_i 。

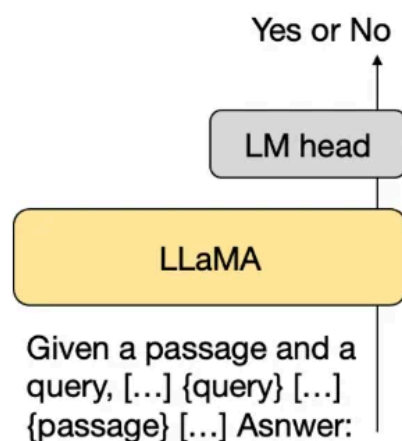
损失函数采用**RankNet损失**，衡量相对段落顺序的正确性。

$$\mathcal{L}_{\text{RankNet}} = \sum_{i=1}^M \sum_{j=1}^M \mathbb{1}_{r_i < r_j} \log(1 + \exp(s_i - s_j))$$

- $\mathbb{1}_{r_i < r_j}$ 是一个指示函数，如果 $r_i < r_j$ ，则为1，否则为0，作用是只考虑 i 对应 passage 排名比 j 靠前的情况。
- $\log(1 + \exp(s_i - s_j))$ 计算两个 passage 相关性得分差异的一种方式，先计算 $s_i - s_j$ 目标训练模型计算出的相关性得分的差值。
 - **BERT-like 模型**：将查询和段落用 [SEP] 标记连接起来，并使用 [CLS] 标记的表示来估计相关性。
 - **GPT-like 模型**：通过生成一个相关性 token 来将查询和段落分类为相关或不相关，相关性得分则定义为相关性 token 的生成概率。



(a) BERT-like Specialized Model



(b) GPT-like Specialized Model

GPT-like 模型的prompt示例如下：

Given a passage and a query, predict whether the passage includes an answer to the query by producing either 'Yes' or 'No'.

Passage: {{passage}}

Query: {{query}}

Does the passage answer the query?

Answer: