为交叉编码器重排序器生成合成文档：ChatGPT 与人类专家的比较研究

Arian Askari
a.askari
@liacs.leidenuniv.nl 莱顿大学
荷兰

Evangelos Kanoulas
e.kanoulas@uva.nl 阿姆斯特丹大学 荷兰

Mohammad Aliannejadi
m.aliannejadi,e.kanoulas
阿姆斯特丹大学 荷兰

Suzan Verberne
s.verberne
@liacs.leidenuniv.nl 莱顿大学 荷兰

arXiv:2305.02320v1 [cs.IR] 3 May 2023

摘要

我们从一个全新的方向探究生成式大语言模型（LLM）在为交叉编码器重排序器生成训练数据方面的效用：生成合成文档，而非合成查询。我们推出了一个新数据集 ChatGPT-RetrievalQA，并比较了使用大语言模型生成数据和人工生成数据微调的模型的效果。生成式大语言模型生成的数据可用于扩充训练数据，尤其适用于标注数据较少的领域。我们基于现有数据集“人类与 ChatGPT 比较语料库”（HC3）构建了 ChatGPT-RetrievalQA；该数据集由公开的问题集及其人类回答和 ChatGPT 回答组成。我们分别使用人工生成数据或 ChatGPT 生成的数据，对一系列交叉编码器重排序器进行微调。在 MS MARCO DEV、TREC DL'19 和 TREC DL'20 上的评估表明，使用 ChatGPT 回答训练的交叉编码器重排序模型，作为零样本重排序器的效果在统计上显著优于使用人类回答训练的模型。在监督设置下，使用人工数据训练的重排序器则优于使用大语言模型生成的数据训练的重排序器。我们的新发现表明，生成式大语言模型在为神经检索模型生成训练数据方面具有巨大潜力。未来仍需进一步研究生成的回答中事实错误信息的影响，并使用开源大语言模型检验这些发现的普适性。我们公开了数据、代码和交叉编码器检查点，以供后续研究使用。[1]

1 引言

GPT-3 [3] 和 GPT-3.5（包括 ChatGPT）等生成式大语言模型（LLM）在摘要生成 [40],机器翻译 [26], 情感分析[32, 36], 检索可解释性 [20],和立场检测 [39]等多种任务中，展现出生成逼真文本的卓越能力。尽管 ChatGPT 能够给出令人印象深刻的答案，但它也无法避免错误或幻觉 [12]。此外，ChatGPT生成的信息来源缺乏透明度，这在法律、医学和科学等领域可能是更大的隐忧，因为这些领域对问责制和可信度有着至关重要的要求 [4, 28, 31]。

与生成模型不同，检索模型会从来源中检索实际（真实）信息，而搜索引擎会提供每个检索结果的来源 [29]。因此，即使已有生成式大语言模型，信息检索（IR）仍然是一项重要应用，尤其是在可靠性至关重要的情况下。生成式大语言模型在信息检索中的一个潜在用途，是为检索模型生成训练数据。生成式大语言模型生成的数据可用于扩充训练数据，尤其适用于标注数据较少的领域。本文提出 ChatGPT-RetrievalQA 数据集，以探讨两个研究问题：**RQ1：** 在有监督和零样本设置下，基于 ChatGPT生成的响应微调的交叉编码器重排序器，其效果与基于人工生成的回答微调的重排序器相比如何?

研究问题2： 使用 ChatGPT 生成相关文档，在特定领域和通用领域中的效果有何不同?

通过回答这些问题，我们旨在阐明使用大语言模型为交叉编码器重排序器进行数据增强的潜力，以及其效果对领域的依赖性。如图1所示，我们的主要实验设置是使用 CEChatGPT[2] 对人工生成的回答进行推理（即重排序任务）。

我们的数据集和分析揭示了使用生成式大语言模型增强检索模型训练数据的优势与局限。

本研究的主要贡献有三个方面：（i）我们发布了 ChatGPT-RetrievalQA 数据集，该数据集基于 HC3 数据集 [13] ，但专门针对全排序和重排序两种设置下的信息检索任务进行设计。该数据集包含 24,322 条查询、26, 882 条由 ChatGPT 生成的响应，以及 58,546 条人工生成的回答。（ii）我们使用人工生成和 ChatGPT 生成的响应对交叉编码器重排序器进行微调，并在监督设置下评估其在我们数据集上的性能。我们还展示了使用 ChatGPT 训练的模型在 MS MARCO 段落集合和 TREC 深度学习赛道上进行零样本评估时的有效性。（iii）我们分析了使用 ChatGPT 训练的交叉编码器在不同领域中的有效性，并发现经过人工数据训练的模型在特定领域任务中略具优势，例如在医学领域。我们的新颖发现凸显了使用生成式大语言模型的潜力。

[1] https://github.com/arian-askari/ChatGPT-RetrievalQA

[2] 我们分别将基于 ChatGPT 生成的回答和人工生成的回答微调的交叉编码器称为 $CE_{ChatGPT}$ 和 CE_{human}。

版本 *1.0*，*2023*年*5*月*3*日，.

表 1：用于评估交叉编码器的各领域训练集、验证集和测试集规模统计。

领域	查询数量 训练集	验证集	测试集
All	16788	606	6928
医学：MedDialog [5]	862	31	355
金融：FiQA [22]	2715	98	1120
Reddit：ELI5 [11]	11809	427	4876
维基百科：OpenQA [38]	820	29	338
维基百科：csai [13]	582	21	239

例如 ChatGPT，用于在信息检索任务中生成高质量响应，以创建训练数据集。

2 相关工作

InPars [2], Promptagator [8],和 InPars-v2 [15]利用大语言模型，根据文档生成合成查询。具体而言，InPars- v2 [15] 使用开源语言模型 GPT-J-6B [33]以及强大的外部重排序器 MonoT5-MSMARCO [24]，筛选出排名前 10k 的高质量合成查询-文档对，以进行数据增强，并在零样本设置下于 BEIR数据集上取得了最先进的结果。相比之下，我们使用 ChatGPT 根据查询生成的文档（段落），这与 InPars-v2 的做法恰好相反。此前的研究尚未探索将针对给定查询生成文档作为数据增强的来源。

我们认为，探索这一反向方向十分重要，因为这使我们能够围绕用户行为和查询日志来扩充训练数据，而不是围绕（静态的）文档集合本身。根据真实搜索引擎用户输入的查询生成合成文档，以此扩充训练数据，可以提高重排序器的效果，增加训练数据的多样性，同时帮助排序器更好地泛化到新的查询。

Guo等人 [13]使用公开的问答数据集（见下文），并将问题输入 ChatGPT 用户界面以生成答案。HC3 数据集旨在从语言学角度比较人工生成的回答与 ChatGPT 的回答，并探索区分 ChatGPT 生成的回答与人工撰写的回答的可能性。HC3 数据集包含来自四个不同领域的问题（即查询）：医学（Medical Dialog[5]）、金融（FiQA [22]）、维基百科（WikiQA [38] 和 Wiki_csai [13]）以及 Reddit（ELI5 [11]）。尽管目前尚无通过生成文档来扩充训练数据的研究，但较新的一项研究 QuerytoDoc [34], 会生成用于查询扩展的文档，该研究超出了信息检索数据增强的范围。此外，近期有多项研究探讨了 ChatGPT 在排序和检索方面的应用，但据我们所知，这些研究都没有关注通过生成相关文档来进行数据增强。近期研究的例子包括 Faggioli 等人[10], 的研究，他们探讨了 ChatGPT 是否可用于生成相关性标签；以及 Sun 等人 [30], 的研究，他们通过向 ChatGPT 提供一个查询和一组候选文档，并要求其重新排序，来评估 ChatGPT 是否擅长搜索。

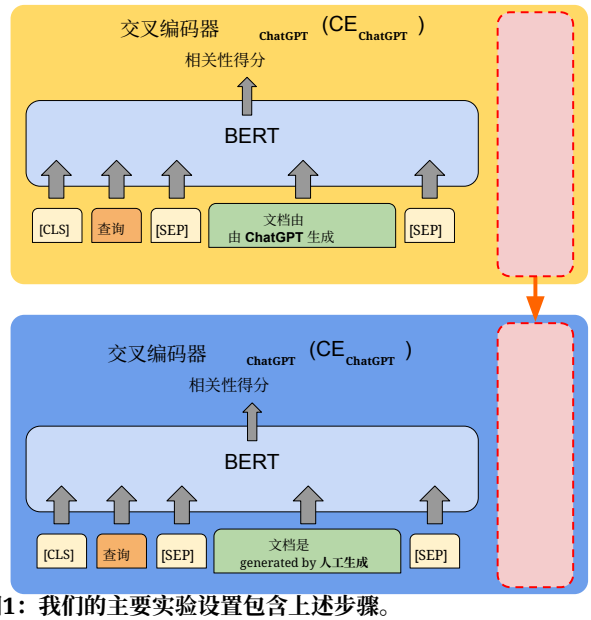

图1：我们的主要实验设置包含上述步骤。

3 数据集

我们的 ChatGPT-RetrievalQA 数据集基于 Guo等人制作的 HC3 数据集 [13], , 其中包含 24,322 条查询、26, 882 条 ChatGPT生成的响应，以及 58, 546 条人工生成的响应。平均每条查询对应一条 ChatGPT生成的响应和 2.4 条人工生成的响应。为了构建用于检索和排序实验的 ChatGPT-RetrievalQA 数据集（与 [13] 中不同的实验设置），我们将数据集文件转换为类似 MSMarco 数据集 [23], 的格式，分别用于全排序和重排序设置。[3] 我们将数据划分为训练集、验证集和测试集。

为便于训练，我们提供了 TSV 格式的训练三元组文件，其中包含文本表示和基于 ID 的表示。每行由“查询、正例段落、负例段落”和“qid、正例 pid、负例 pid”组成。我们将 ChatGPT 或人工生成的实际回答视为相关答案，并参照 MS MARCO，为每个查询随机采样 1000 个负例答案。此外，我们还提供了每个查询经 BM25 排序后的前 1000 篇文档，以便开展重排序研究。表1展示了各个领域的训练集、验证集和测试集规模。

4 方法

4.1 第一阶段排序器：BM25

词汇检索器通过词语重叠来计算文档与查询之间的相关性得分。过去已有多种词汇方法被提出，例如向量空间模型、Okapi B M25 [27], 和查询似然。我们使用 BM25 作为第一阶段排序器，因为它广受欢迎且效果显著。BM25 根据以下统计数据计算查询–文档对的得分。

[3]这样便于重复利用 MS MARCO 上现有的脚本。

表 2：比较在监督设置和零样本设置下，基于人工生成和 ChatGPT 响应微调的交叉编码器重排序器的效果。† 表示，对于该数据集，在所有交叉编码器重排序器和该数据集上的 BM25 中，某个交叉编码器取得了具有统计显著性的改进。统计显著性通过配对 t 检验（$p < 0.05$）并采用 Bonferroni 校正进行多重检验来衡量。MAP、NDCG 和 MRR 的截断值分别为 1000、10 和 10。

	零样本设置									监督设置		
模型	TREC DL'19			TREC DL'20			MS MARCO DEV			ChatGPT-RetrievalQA（**我们的方法**）		
	平均精度均值	归一化折损累计增益	平均倒数排名	平均精度均值	归一化折损累计增益	平均倒数排名	平均精度均值	归一化折损累计增益	平均倒数排名	平均精度均值		MRR
BM25	**.377**	.506	.858	.286	.480	.819	.195	.234	.187	.143	.184	.240
MiniLM人工生成	.326	.451	.833	.269	.376	.913	.130	.155	.118	**.310**†	**.384**†	**.460**†
MiniLMChatGPT	.342†	**.510**†	.903	**.344**†	**.539**†	**.978**†	**.226**†	**.267**†	**.218**†	.294	.362	.444
TinyBERThuman	.294	.360	.741	.277	.364	.791	.128	.154	.116	.244	.310	.367
TinyBERTChatGPT	.328	.488	**.942**†	.303	.460	.972	.194	.231	.185	.231	.291	.358

它们之间重叠的词：

$$s_{lex}(q,d) = BM25(q,d) = \sum_{t \in q \cap d} rsj_t \cdot \frac{tf_{t,d}}{tf_{t,d} + k_1\{(1-b) + b\frac{|d|}{l}\}} \quad (1)$$

其中，t是一个词项，$tf_{t,\ d}$是 t在文档 d中的词频，rsj_t是 t的 Robertson-Spärck Jones 权重 [27]，l是平均文档长度。k_1 和 b 是参数。

4.2 交叉编码器重排序器

在重排序设置中，采用交叉编码器架构的预训练 Transformer 模型，常见做法是将查询和段落的输入序列连接起来。这种方法称为 MonoBERT 或 CECAT，如图1所示，并已应用于多项研究中。在 CECAT 中，查询词序列 $q_1 : q_m$ 和段落词序列 $p_1 : p_n$ 通过 [SEP]标记连接起来，CECAT 的排序模型使用单个线性层 Ws 计算交叉编码器（CE）生成的 [CLS]标记表示的分数：

$$CE_{CAT}(q_{1:m}, p_{1:n}) = CE([CLS]\ q\ [SEP]\ p\ [SEP]) * W_s \quad (2)$$

我们使用 CECAT 作为交叉编码器重排序器，重排序深度为 1000。在我们的实验中，CEChatGPT 和 CEhuman 均遵循上述设计。

5 实验设计

评估设置。 我们在 MS MARCO 段落集合 [23] 以及两个 TREC 深度学习赛道（TREC-DL'19 和 DL'20） [6, 7]上开展零样本评估实验。所使用的评估指标包括 MAP@1000、NDCG@10 和 MRR@10，这些指标是此类数据集的标准指标，以便使我们的结果能够与此前发表的研究和即将开展的研究进行比较 [6, 7]。MS MARCO 段落数据集包含约 8.8 百万个段落（平均长度：73.1 个单词）以及约 100 万个自然语言查询（平均长度：7.5 个单词），并被广泛用于训练用于排序的深度语言模型。遵循此前关于 MS MARCO 的研究 [17, 19, 21, 42, 43], ，我们仅使用开发集（~ $7k$ 个查询）进行实证评估。TREC DL'19 和 DL'20 集合共享 MS MARCO 的段落语料库，分别包含 43 个和 54 个查询，并且每个查询对应的相关文档数量更多。我们在 ChatGPT-RetrievalQA 的测试集上，在监督设置下测量相同的指标。平均长度为

表3：比较 CEC 和 CEH 在不同领域的无监督设置下的效果，其中 CE、C 和 H 分别指 MiniLM、人工生成和 ChatGPT。OpenQA 和 Wiki_csai 数据集属于维基百科领域。

领域	模型	MAP@1K	NDCG@10	MRR@10
All	CE_H	**.310**	**.384**	**.460**
	CE_C	.294	.362	.444
医学 [5]	CE_H	**.397**	**.419**	**.395**
	CE_C	.379	.400	.377
金融 [22]	CE_H	**.257**	**.399**	**.251**
	CE_C	.250	.368	.245
Reddit [11]	CE_H	**.323**	**.418**	**.543**
	CE_C	.302	.391	.522
OpenQA [38]	CE_H	.322	**.345**	.320
	CE_C	**.331**	.341	**.328**
维基_csai [13]	CE_H	.149	.152	.135
	CE_C	**.163**	**.159**	**.144**

人工回答为 142.5 个单词，而 ChatGPT-RetrievalQA 数据集中的 ChatGPT 回答为 198.1 个单词。

训练配置。 我们使用 Huggingface 库 [37],和 PyTorch [25] 进行交叉编码器重排序的训练和推理。遵循先前的研究 [14],，对于所有交叉编码器层，无论训练的层数多少，我们都使用学习率为 $7 * 10^{-6}$ 的 Adam [18]优化器。训练批次大小设为 32。对于所有交叉编码器重排序器，我们使用交叉熵损失 [41]。对于使用 BM25 进行的词汇检索，我们采用 Elasticsearch [9]的相似度函数。遵循先前的研究 [1, 14]，我们将查询长度上限设为 30 个词元，段落长度上限设为 200 个词元。

6 结果

6.1 主要结果（RQ1）

表 2 展示了 CEhuman 和 CEChatGPT 的效果比较。请注意，对于这两个模型，我们在推理期间评估它们在监督或零样本设置下检索人工生成的 响应的效果。由于 MiniLM（12 层版本） [35] 的实验结果具有竞争力，

与 BERT 重排序器相比， [1]其模型规模仅为三分之一，速度则快六倍。此外，我们还使用 TinyBERT（2 层版本） [16] 进行了实验，以评估结果的普适性。

在**监督设置**中，我们在 ChatGPT-RetrievalQA 数据集的测试集查询和人工文档上进行评估，MiniLMhuman 的表现显著优于其他所有交叉编码器重排序器。尽管其表现低于使用人工数据训练的模型，MiniLMChatGP

以及 TinyBERTChatGPT 在这一设置中仍以较大幅度显著优于强基线 [2],BM25 [27], 。

在**零样本设置**中，MiniLMChatGPT 在 TREC DL'20 和 MS MARCO DEV 上始终显著优于其他交叉编码器重排序器，包括 MiniLMhuman 和 BM25。然而，在 TREC DL'19 上，BM25 在 MAP@1000 指标上取得最高效果，MiniLMChatGPT 在 NDCG@10 指标上表现最佳，而 TinyBERTChatGPT 在 MRR@10 指标上表现最佳。总体而言，我们可以看到，与使用人工生成的回答进行微调的模型相比，使用 ChatGPT生成的响应进行微调的模型在零样本设置中的效果显著更好。

6.2 领域级重排序器效果（RQ2）

表 3 展示了 MiniLMhuman 和 MiniLMChatGPT 重排序器在监督设置下的效果，即在我们数据集的测试集上、涵盖医学、金融、Reddit 和维基百科等所有领域的表现。总体而言，结果表明，除维基百科领域外，MiniLMhuman 在所有领域的效果都优于 MiniLMChatGPT。不过，两者的效果差异较小，且 MiniLMChatGPT 仍能达到合理的效果。在金融领域，与其他领域相比，MiniLMhuman 和 MiniLMChatGPT 的效果都相对较低。在维基百科领域，MiniLMhuman 和 MiniLMChatGPT 的效果水平较为接近。在医学领域，CEhuman 的效果最高。总体而言，这些结果表明，MiniLMhuman 在监督设置下表现更佳，尤其是在医学等特定领域，尽管性能差异较小。

7 讨论

数据重叠。 值得注意的是，在监督设置中，CEhuman重排序器的训练和测试所使用的文档集合是共享的。因此，一些文档可能会在训练和评估期间都被看到。这种设置在使用人工评估数据时非常常见，与MS MARCO数据集类似 [23]。在监督设置中，共享文档集合可能是CEhuman重排序器的一个潜在优势，因为模型可能已经在训练期间见过其中一些文档。为了进一步研究这一假设，未来工作可以探索一种不同的设置，在该设置中，训练集和测试集之间的文档集合完全分离。

BM25 的有效性。表4分析了 BM25 在训练集和测试集中对人工生成的回答和 ChatGPT生成的响应的有效性。与 ChatGPT 生成的响应相比，BM25 对人工生成的回答的效果较差，这体现在训练集和测试集的所有指标得分都较低。我们在验证集上也观察到了相同的模式。这些结果表明，

表4：分析 BM25 在训练集、验证集和测试集上对人类/ChatGPT 回答的效果。

拆分	来源	MAP@1K	NDCG@10	Recall@1K
Test	人工生成	.143	.184	.520
	ChatGPT	.370	.396	.898
训练集	人工生成	.158	.202	.560
	ChatGPT	.413	.443	.903

T

表5：分析 CEsChatGPT 在训练集的已见查询和人工生成文档集合的未见文档上的有效性。

模型	MAP@1K	归一化折损累计增益@10	MRR@10
MiniLMChatGPT	.318	.388	.510
TinyBERTChatGPT	.254	.318	.420

对于 BM25 来说，检索人工生成的回答比检索 ChatGPT生成的响应更具挑战性。这可能与下文讨论的词汇重叠有关。

无标签查询。表5中，我们考察了真实世界搜索引擎中的一种常见情形：查询日志和一组人工生成的文档均已具备，但查询日志中的部分或全部查询没有任何经过标注的文档。为模拟和分析这一情况，我们在训练集中的已见查询以及人工生成文档集合中的未见文档上评估 CEChatGPT。表5显示，CEChatGPT 排序器在这种情形下相当有效。尤其是在相同设置下，它们比 BM25 更有效：MiniLMChatGPT 的 NDCG@10 为 0.388，而 BM25 为 0.202（见表4第三行）。这表明，在没有查询的相关性判断文档时，可以利用生成式大语言模型扩充训练数据，以微调模型，使其能够根据查询日志对语料库中来源可靠的人工生成文档进行有效的重新排序。

词汇重叠。我们的数据分析表明，与查询相比，ChatGPT生成的响应与查询的词汇重叠程度略高于人工生成的回答。ChatGPT生成的响应中出现的查询词平均占查询词总数的34.6%，而人工生成的回答为25.5%。与人类回答相比，ChatGPT的第一四分位数、中位数和第三四分位数平均也高出7个百分点。我们推测，与人工生成的回答相比，ChatGPT生成的响应词汇重叠程度更高，是因为ChatGPT经常在回答中重复问题或查询，而且其生成的响应通常比人工生成的回答更长，从而增加了与查询包含相同词语的可能性。值得注意的是，词汇重叠并非衡量响应质量、以便对交叉编码器进行有效微调的最佳指标，因为有时词汇重叠较低的响应仍然相关且信息丰富，尤其是在问答任务中。

8 结论

本文分析了在人工生成的回答与 ChatGPT生成的响应上微调交叉编码器的效果。结果表明，在

零样本设置下，交叉编码器ChatGPT 的效果优于人工交叉编码器；而在监督设置下，MiniLM 人工生成版的效果略好，这一结论在不同领域中均成立。此外，我们发现，BM25 在人工生成的回答上的效果不如在 ChatGPT生成的响应上的效果，这表明，与ChatGPT生成的响应相比，人工生成的回答更难与查询匹配。总体而言，我们的研究结果表明，至少基于本研究的数据集和实验，ChatGPT生成的响应比人工生成的回答更适合用于训练高效的零样本重排序器；同时也凸显了利用生成式大语言模型生成有效且实用的回答，以创建自然语言处理任务训练数据集的潜力。对于特定领域任务而言，本研究尤其有价值，因为在这类任务中，将大语言模型生成的输出直接作为对用户查询的回答可能存在风险。我们的结果证实，即使针对特定领域查询，也可以通过使用ChatGPT生成的响应进行训练，得到有效的交叉编码器重排序器。后续研究还需确定生成的回答中事实错误信息的影响，并检验我们的发现能否推广至开源大语言模型。

参考文献

[1] Arian Askari、Amin Abolghasemi、Gabriella Pasi、Wessel Kraaij 和 Suzan Ver- berne。2023。将 BM25 分数作为文本注入可提升基于 BERT 的重排序器。载于信息检索进展。Springer Nature Switzerland，Cham，66–83。[2] Luiz Bonifacio、Hugo Abonizio、Marzieh Fadaee 和 Rodrigo Nogueira。2022。Inpars：使用大语言模型进行信息检索数据增强。*arXiv* 预印本 *arXiv:2202.05144* (2022)。[3] Tom Brown、Benjamin Mann、Nick Ryder、Melanie Subbiah、Jared D Kaplan、Prafulla Dhariwal、Arvind Neelakantan、Pranav Shyam、Girish Sastry、Amanda Askell 等。2020。语言模型是少样本学习者。神经信息处理系统进展33 (2020)，1877–1901。[4] Marco Cascella、Jonathan Montomoli、Valentina Bellini 和 Elena Bignami。2023。评估 ChatGPT 在医疗保健领域的可行性：对多个临床和研究场景的分析。《医学系统杂志》47, 1 (2023)，1–5。[5] Shu Chen、Zeqian Ju、Xiangyu Dong、Hongchao Fang、Sicheng Wang、Yue Yang、Jiaqi Zeng、Ruisi Zhang、Ruoyu Zhang、Meng Zhou 等。2020。MedDialog：大规模医疗对话数据集。*arXiv* 预印本 *arXiv:2004.03329* (2020)。[6] Nick Craswell、Bhaskar Mitra、Emine Yilmaz 和 Daniel Campos。2021。TREC 2020 深度学习赛道概览。*arXiv* 预印本 *arXiv:2102.07662* (2021)。[7] Nick Craswell、Bhaskar Mitra、Emine Yilmaz、Daniel Campos 和 Ellen M Voorhees。2020。TREC 2019 深度学习赛道概览。*arXiv* 预印本 *arXiv:2003.07820* (2020)。[8] Zhuyun Dai、Vincent Y Zhao、Ji Ma、Yi Luan、Jianmo Ni、Jing Lu、Anton Bakalov、Kelvin Guu、Keith B Hall 和 Ming-Wei Chang。2022。Promptagator：基于 8 个示例的少样本密集检索。*arXiv* 预印本 *arXiv:2209.11755* (2022)。[9] BV Elasticsearch。2023。Elasticsearch。软件*]*, 版本 6, 1 (2023)。[10] Guglielmo Faggioli、Laura Dietz、Charles Clarke、Gianluca Demartini、Matthias Hagen、Claudia Hauff、Noriko Kando、Evangelos Kanoulas、Martin Potthast、Benno Stein 等。2023。关于大语言模型用于相关性判断的观点。*arXiv* 预印本 *arXiv:2304.09161* (2023)。[11] Angela Fan、Yacine Jernite、Ethan Perez、David Grangier、Jason Weston 和 Michael Auli。2019。ELI5：长篇问答。*arXiv* 预印本 *arXiv:1907.09190*(2019)。[12] Josh A Goldstein、Girish Sastry、Micah Musser、Renee DiResta、Matthew Gentzel 和 Katerina Sedova。2023。生成式语言模型与自动化影响力行动：新兴威胁与潜在缓解措施。*arXiv* 预印本 *arXiv:2301.04246* (2023)。[13] Biyang Guo、Xin Zhang、Ziyuan Wang、Minqi Jiang、Jinran Nie、Yuxuan Ding、Jianwei Yue 和 Yupeng Wu。2023。ChatGPT 与人类专家有多接近？比较语料库、评估与检测。*arXiv* 预印本 *arXiv:2301.07597* (2023)。[14] Sebastian Hofstätter、Sophia Althammer、Michael Schröder、Mete Sertkan 和 Allan Hanbury。2020。通过跨架构知识蒸馏改进高效神经排序模型。*arXiv* 预印本 *arXiv:2010.02666* (2020)。[15] Vitor Jeronymo、Luiz Bonifacio、Hugo Abonizio、Marzieh Fadaee、Roberto Lotufo、Jakub Zavrel 和 Rodrigo Nogueira。2023。InPars-v2：将大语言模型用作信息检索的高效数据集生成器。*arXiv* 预印本 *arXiv:2301.01820* (2023)。[16] Xiaoqi Jiao、Yichun Yin、Lifeng Shang、Xin Jiang、Xiao Chen、Linlin Li、Fang Wang 和 Qun Liu。2020。TinyBERT：蒸馏 BERT 以处理自然语言理解。在《计算语言学协会发现：EMNLP 2020》中。计算语言学协会，线上，4163–4174。https://doi.org/10 18653 1 2020 372 . /v / .findings-emnlp. Omar Khattab [17] 和 Matei Zaharia。. Colbert：2020 通过在 BERT 上进行上下文化后期交互，实现高效且有效的段落检索。载于《第43届国际ACM SIGIR信息检索研究与发展会议论文集》。39–48。[18] Diederik P Kingma 和 Jimmy Ba。2014。Adam：一种随机优化方法。arXiv 预印本 arXiv:1412.6980（2014）。[19] Jimmy Lin、Rodrigo Nogueira 和 Andrew Yates。2021。用于文本排序的预训练变换器：BERT及其他模型。《人类语言技术综合讲座》14，4（2021），1–325。[20] Michael Llordes、Debasis Ganguly、Sumit Bhatia 和 Chirag Agarwal。2023。像向我解释 BM25 一样解释：用稀疏近似解读稠密模型的排序列表。arXiv 预印本 arXiv:2304.12631（2023）。[21] Sean MacAvaney、Franco Maria Nardini、Raffaele Perego、Nicola Tonellotto、Nazli Goharian 和 Ophir Frieder。2020。通过重要性预测并结合上下文化进行扩展。载于《第43届国际ACM SIGIR信息检索研究与发展会议论文集》。1573–1576。[22] Macedo Maia、Siegfried Handschuh、André Freitas、Brian Davis、Ross McDermott、Manel Zarrouk 和 Alexandra Balahur。2018。Www'18开放挑战赛：金融观点挖掘与问答。载于《2018年万维网大会配套论文集》。1941–1942。[23] Tri Nguyen、Mir Rosenberg、Xia Song、Jianfeng Gao、Saurabh Tiwary、Rangan Majumder 和 Li Deng。2016。MS MARCO：人工生成的机器阅读理解数据集。载于 CoCo@ NIPs。[24] Rodrigo Nogueira、Zhiying Jiang 和 Jimmy Lin。2020。使用预训练序列到序列模型进行文档排序。arXiv 预印本 arXiv:2003.06713（2020）。[25] Adam Paszke、Sa m Gross、Soumith Chintala、Gregory Chanan、Edward Yang、Zachary DeVito、Zeming Lin、Alban Desmaison、Luca Antiga 和 Adam Lerer。2017。PyTorch中的自动微分。（2017）。[26] Keqin Peng、Liang Ding、Qihuang Zhong、Li Shen、Xuebo Liu、Min Zhang、Yuanxin Ouyang 和 Dacheng Tao。2023。充分发挥 ChatGPT 在机器翻译中的作用。arXiv 预印本 arXiv:2303.13780（2023）。[27] Stephen E Robertson 和 Steve Walker。1994。概率加权检索中二泊松模型的一些简单有效近似。载于 SIGIR'94。Springer，232–241。[28] Malik Sallam、Nesreen Salim、Muna Barakat 和 Alaa Al-Tammemi。2023。ChatGPT在医学、牙科、药学和公共卫生教育中的应用：一项突出优势与局限性的描述性研究。Narra J 3，1（2023），e103–e103。[29] Mark Sanderson 和 W Bruce Croft。2012。信息检索研究史。IEEE论文集 100，百年特刊（2012），1444–1451。[30] Weiwei Sun、Lingyong Yan、Xinyu Ma、Pengjie Ren、Dawei Yin 和 Zhaochun Ren。2023。ChatGPT擅长搜索吗？将大语言模型作为重排序智能体进行研究。arXiv 预印本 arXiv:2304.09542（2023）。[31] Zhongxiang Sun。2023。关于从法律方面看待大语言模型的简短综述。arXiv 预印本 arXiv:2303.09136（2023）。[32] Teo Susnjak。2023。使用 BERT 和 ChatGPT 对科学文献中的莱姆病进行情感分析。arXiv 预印本 arXiv:2302.06474（2023）。[33] Ben Wang 和 Aran Komatsuzaki。2021。GPT-J-6B：一个拥有60亿参数的自回归语言模型。https://github.com/kingoflolz/mesh-transformer-jax。[34] Liang Wang、Nan Yang 和 Furu Wei。2023。Query2doc：使用大语言模型进行查询扩展。arXiv: 2303 07678 [ ] . cs.IR Wenhui Wang、[35] Furu Wei、Li Dong、Hangbo Bao、Nan Yang 和 Ming Zhou。2020。MiniLM：用于预训练变换器任务无关压缩的深度自注意力蒸馏。《神经信息处理系统进展》33（2020），5776–5788。[36] Zengzhi Wang、Qiming Xie、Zixiang Ding、Yi Feng 和 Rui Xia。2023。ChatGPT是优秀的情感分析器吗？初步研究。arXiv 预印本 arXiv:2304.04339（2023）。[37] Thomas Wolf、Lysandre Debut、Victor Sanh、Julien Chaumond、Clement Delangue、Anthony Moi、Pierric Cistac、Tim Rault、Rémi Louf、Morgan Funtowicz 等。2019。Hugging Face的Transformers：最先进的自然语言处理。arXiv 预印本 arXiv:1910.03771（2019）。[38] Yi Yang、Wen-tau Yih 和 Christopher Meek。2015。WikiQA：开放域问答挑战数据集。载于《2015年自然语言处理实证方法会议论文集》。2013–2018。[39] Bowen Zhang、Daijun Ding 和 Liwen Jing。2022。ChatGPT发布后立场检测技术将如何演进？arXiv 预印本 arXiv:2212.14548（2022）。[40] Haopeng Zhang、Xiao Liu 和 Jiawei Zhang。2023。通过 ChatGPT 进行抽取式摘要，以生成忠实摘要。arXiv 预印本 arXiv:2304.04193（2023）。[41] Zhilu Zhang 和 Mert Sabuncu。2018。用于以噪声标签训练深度神经网络的广义交叉熵损失。《神经信息处理系统进展》31（2018）。

[42] Shengyao Zhuang、Hang Li 和 Guido Zuccon。2021 年。用于信息检索的深度查询似然模型。载于欧洲信息检索会议。Springer，463–470 页。

[43] Shengyao Zhuang 和 Guido Zuccon。2021。TILDE：用于段落重排序的词项独立似然模型。在第 *44* 届国际 *ACM SIGIR* 信息检索研究与发展会议论文集中。1483–1492。