arXiv:2301.05508v1 [cs.IR] 13 Jan 2023

文档与段落检索的研究发现能否推广到对话响应检索?

Gustavo Penha 和 Claudia Hauff

代尔夫特理工大学{
g.penha-1,c.hauff}@tudelft.nl

摘要。近期提出了多种基于学习的稀疏和稠密检索方法，并已证明它们在段落检索和文档检索等任务中有效。本文通过一项可复现性研究，分析已获得的经验能否推广到对话响应检索这一任务；对于日益流行的对话式搜索领域而言，这是一项重要任务。段落和文档检索中的文档通常比查询长，而在对话响应排序中，查询（对话上下文）往往比文档（响应）长。此外，对话具有特定结构，即由不同用户发出的多个话语。考虑到这些差异，我们在此评估先前研究中的以下主要发现有多大程度的普适性：（F1）查询扩展优于无扩展基线；（F2）文档扩展优于无扩展基线；（F3）零样本稠密检索的表现不及稀疏基线；（F4）稠密检索优于稀疏基线；（F5）训练稠密模型时，困难负采样优于随机采样。我们的实验[1]——基于三个不同的信息寻求对话数据集——表明，五项发现中有四项（F2–F5）可推广到我们的领域。

1 引言

对话式搜索旨在通过自然语言交互进行混合主动式对话，而非借助传统的搜索引擎结果页，创建能够满足信息需求的智能体。对话式搜索的一种常见方法是基于检索的方法 [3]：给定正在进行的对话以及大型历史对话语料库，从语料库中检索最合适的响应 [45,48,28,47,11]。由于 BERT 等经过大量预训练的基于 Transformer 的语言模型效果出色 [4], ，它们已成为对话响应重新排序的主流方法 [28,43,53,8,42]。

对话响应重排序最常见的评估流程，是对一组有限的 n 候选响应（包括

[1] https://github.com/Guzpenha/transformer_rankers/tree/full_排序_检索_对话。

真实回复）进行重排序，然后统计前 K 个位置中找到的相关响应数量——$Recall_n$ @K [52]。由于可用响应的全集通常远大于这样一组候选项，因此这种设置实际上是一个选择问题：我们必须从少数几个选项中选出正确的响应。这种评估忽略了第一阶段检索步骤，而该步骤负责检索一组 n 响应，以供后续重排序。如果第一阶段模型（例如 BM25）未能检索到相关响应，整个流程就会失败。

由于关于第一阶段检索步骤的研究不足，我们希望通过这项可复现性研究，探讨文档与段落检索任务中积累的大量知识能否推广到对话领域。与文档通常比查询更长的文档与段落检索不同，在对话响应检索中，查询（对话上下文）往往比文档（响应）更长。另一个重要区别在于对话所带来的结构，如表 1 所示。

表 1. 段落检索与对话响应检索的比较。在 §3 中，我们定义了 *First-stage Retrieval for Dialogues* 任务。颜色代表 信息寻求者 和 信息提供者。 p^+/r^+ 是相关段落/响应。

	段落检索	对话的第一阶段检索
输入	查询 q	对话上下文 $\mathcal{U} = \{u^1, u^2, ..., u^\tau\}$
示例	q: *what is theraderm used for*	u^1: *I want a firewall that will protect me but more of that to monitor any connection in or out of my mac [...]* u^2: *{url} allows you to map joypad buttons to keyboard keys and [...]* u^3: *Do the diagonals for the analog stick work correctly for you? [...]*
输出	段落排序列表	响应排序列表
示例	p^+: *Thera-Derm Lotion is used as a moisturizer to treat or prevent dry, rough, scaly, [...]*	r^+: *In the "Others" tab, try [...]*

鉴于领域之间存在差异，我们在三个信息寻求数据集和 170 万个查询上，通过实证验证了段落和文档检索文献中与最先进的稀疏和稠密检索模型相关的五项发现（F1 至 F5）的泛化能力。我们之所以选择这五项发现，是因为它们在先前研究中影响重大（参见 §2）。结果表明，先前的五项发现中有四项确实能够泛化到我们的领域：

[2]虽然对于大多数基准测试，[52] 我们只有 10–100 个候选项，但使用 PolyAI 的 Reddit 数据（https://github.com/PolyAI-LDN/conversational-datasets）构建的实际系统，需要从 37 亿条响应中进行检索。

F1 ✗[3] 对话上下文（即查询）扩展优于无扩展基线 [1,18,49,21]。**F2** ✓ 响应（即文档）扩展优于无扩展基线[25,21,19]，前提是扩展模型经过训练，能够生成最近的上下文（对话中的最后一句话[4]），而非较早的上下文（所有话语）。**F3** ✓ 零样本设置下的稠密检索[5] 表现不如稀疏基线[34,41]，但若先利用大量域外数据进行中间训练，则是例外。**F4** ✓ 能够使用目标数据的稠密检索[6] 优于稀疏基线 [7,15,34]，前提是在目标数据微调之前，先使用域外数据进行中间训练。**F5** ✓ 如果使用去噪技术减少假负例样本的数量，更难的负采样技术可带来效果提升 [46,51] 。

我们的结果表明，大多数研究发现都适用于对话响应检索。因此，一个颇有前景的未来方向是从其他领域已取得成功的模型入手——这些领域拥有更多数据集和既有研究——并探讨如何对其进行调整和改进，以用于基于检索的对话式搜索。

2 相关工作

本节首先讨论对话式搜索中基于检索的系统的当前研究，随后回顾段落和文档检索领域中（无）监督式稀疏检索与稠密检索的主要研究发现。

2.1 对话响应的排序与检索

早期用于响应重排序的神经模型采用卷积神经网络和长短期记忆网络等架构，以单轮方式匹配拼接后的对话上下文表示与响应表示 [23,14]。研究者也探索了更复杂的神经网络架构，用于将每个话语与响应进行匹配 [54,9,22]。Nogueira 和 Cho [24] 首先证明，经过大量预训练的语言模型（如 BERT）在重排序任务中效果显著。此类模型迅速成为信息检索（IR）中重排序的主流方法 [21] ，后来也被证明适用于对话中的响应重排序 [42,28]。

相比之下，对话响应的第一阶段检索受到的关注相对较少 [29]。Lan 等人 [17] 和 Tao 等人 [38] 表明，基于 BERT 的稠密检索模型在对话响应的第一阶段检索中优于 BM25。他们研究的一个局限是，强大的稀疏检索

[3] ✗ 表示该发现不适用于我们的领域，而 ✓ 表示该发现适用于我们的领域，后面跟着 *necessary condition or exception*。 [4] 例如，在表1中，最后一句话是 u^3。 [5] 零样本模型是指无法访问目标数据的模型，参见表2。 [6] 目标数据是指来自与评估数据集相同分布（即同一数据集）的数据。

基线在其他检索任务中已被证明有效，例如结合对话上下文扩展的 BM25 [25] 或结合响应扩展的 BM25 [49], ，但未用于稠密检索。我们在此进行了此类比较，并检验了五项重要发现；此前关于对话响应第一阶段检索的研究尚未对这些发现进行评估。

2.2 段落和文档检索的稠密与稀疏模型

F1 的上下文：检索模型可以从两个维度进行分类：有监督与无监督，以及稠密与稀疏表示 [19]。诸如 BM25 [35] 这样的无监督稀疏表示模型，会用维度等于语料库词汇表大小的稀疏向量表示每篇文档和每个查询；由于某些词项并未出现，向量中会有许多权重为零。由于每个词项的权重完全基于词项统计数据，因此这类方法被视为无监督方法。这类方法容易受到词汇不匹配问题 [6],的影响，因为它们不考虑语义匹配。解决这一问题的一种方法是使用查询扩展方法。RM3 [1] 是一种有竞争力的 [49] 查询扩展技术，它利用伪相关反馈向查询中添加新词项，然后使用修改后的查询执行最后一次检索。

F2 的上下文：有监督稀疏检索模型可以通过将词项权重从语料库统计数据改为学习得到的值，利用基于 Transformer 的语言模型的有效性。使用学习式模型进行文档扩展，可视为一种学习式稀疏检索方法 [19]。其核心思想是创建包含扩展词项的伪文档，并在检索时用这些伪文档替代原文档。Doc2query [25] 是一种表现出色的有监督稀疏检索基线，它使用语言模型预测可能用于查找某篇文档的查询。该模型的预测结果用于创建扩展后的伪文档。

F3 和 F4 的背景 监督式稠密检索模型[7], ，例如 ANCE [46]和 coCodenser [7], ，会将查询和文档表示为一个较小的固定长度空间中的向量，例如 768 维空间中的向量。无法使用目标数据进行训练的稠密检索模型——这种情况称为零样本场景——表现不如稀疏方法（F3）。例如，BEIR 基准 [41] 显示，在零样本场景下，18 个数据集中有 9 至 18 个（视模型而定）数据集的检索效果是 BM25 优于稠密检索。相比之下，在能够使用足够的目标数据进行监督时，稠密检索模型的表现已被证明始终优于强大的稀疏基线 [7,15,34] （F4）。

[7] 还可以区分交叉编码器和双编码器：前者将查询和文档联合编码，而后者则分别编码 [40]。由于效率低下，交叉编码器会用于重排序步骤，因此不是我们的关注重点。

F5 的背景 为了训练神经排序模型，需要少量负例（即不相关）候选项，因为将集合中的其他每篇文档都用作某个查询的负样本，成本高得难以承受。随机选择负样本的一个局限是，这些样本可能过于容易区分，排序模型很容易将其与相关项区分开来；而对于难度更高的负文档，模型可能仍难以处理。因此，研究表明，在段落和文档检索中，困难负采样的表现优于随机采样 [46,36,51]。

3 对话的第一阶段检索

本节首先介绍响应第一阶段检索问题，然后介绍我们希望从稀疏方法和稠密方法中复现的发现。

问题定义 对话响应第一阶段检索任务旨在给定对话上下文后，从整个集合中检索出最佳响应。形式上，令 $\mathcal{D} = \{(\mathcal{U}_i, \mathcal{R}_i, \mathcal{Y}_i)\}_{i=1}^{M}$ 为一个数据集，其中包含M个三元组：对话上下文、响应候选项和响应相关性标签。对话上下文 $\mathcal{U}_i$ 由对话中第 τ轮之前的话语 $\{u^1, u^2, ..., u^\tau\}$组成。候选响应 $\mathcal{R}_i = \{r^1, r^2, ..., r^n\}$ 要么是真实响应 r^+ ，要么是负采样候选项 r^-，由相关性标签 $\mathcal{Y}_i = \{y^1, y^2, ..., y^n\}$ 指示。在以往的工作中，候选项数量有限，通常为 $n = 10$ [29]。我们在此复现的发现来自段落和文档检索任务，在这些任务中，需要检索的文档或段落数量没有限制。因此，在我们的所有第一阶段检索任务实验中， n 设为语料库中响应总集合的大小。真实响应的数量为一个，即对话数据中观察到的响应。接下来需要学习一个排序函数$f(.)$，根据预测的相关性分数 $f(\mathcal{U}, r)$，从整个响应语料库 $\mathcal{R}_i$中生成排序列表。

F1：无监督稀疏检索 我们采用经典检索方法，其中最常用的基线是 BM25。稀疏检索的局限之一是词汇不匹配问题。扩展技术可以通过在对话上下文和响应中添加新词来克服这一问题。因此，我们在此将一种查询扩展技术迁移到对话领域，并使用 RM3 [1], 进行对话上下文扩展。RM3 [1], 是一种有竞争力的无监督方法，它假设稀疏检索模型排名靠前的响应是相关的。系统从这些伪相关响应中选取词语，创建扩展后的对话上下文，随后由稀疏检索方法使用该上下文对最终响应列表进行排序。我们验证的第一个发现是 RM3 在对话领域中的有效性。

F2：学习式稀疏检索 另一种做法是使用学习式方法扩展集合中的响应。为此，我们将 doc2query [25] “迁移”到我们的领域，得到 resp2ctxt。具体而言，我们针对从真实回复 r_i^+ 生成对话上下文 $\mathcal{U}_i$ 这一任务，对生成式 Transformer 模型 G 进行微调。随后使用该模型为集合中的所有响应生成扩展内容，$r^i = concat(r^i, G(r^i))$。这些扩展内容会附加到响应中，并再次对集合建立索引——稀疏检索方法本身并未修改，也就是说，我们仍然使用 BM25。这种方法（我们称之为 resp2ctxt）带来了两方面的改进：术语重新加权（添加文档中已有的术语）以及解决词汇不匹配问题（添加新术语）。我们验证的第二个发现是 doc2query 在对话领域中的有效性。

与段落检索和文档检索不同，后两者的查询比文档短，而在对话响应检索中，查询比文档长[8]。这对生成模型来说是一项挑战，因为生成较长的文本片段比生成较短的文本片段更困难，出错的可能性也更大。鉴于此，我们还探索了 resp2ctxt 的一个修改版本，旨在只生成对话上下文中的最后一句话：resp2ctxt$_{lu}$。该模型经过训练，可以从 r_i^+ 生成 u^τ，而不是尝试生成整个话语 $\mathcal{U}_i = \{u^1, u^2, ..., u^\tau\}$。其基本前提是，对话中最重要的话语是最后一句；如果 resp2ctxt$_{lu}$能正确生成这句话，稀疏检索方法就能够从集合中找到正确的响应。

F3：零样本稠密检索 我们采用能够在稠密嵌入空间中分别表示对话上下文和响应的方法。随后，根据响应与对话上下文的相似度对响应进行排序。我们在此使用预训练语言 Transformer 模型，例如 BERT [4] 和 MP-Net [37], ，来获得对话上下文和响应的此类表示。这种方法通常称为双编码器模型 [21] ，是一类有效的模型[9]。零样本模型是指未在目标数据上训练的模型。目标数据是指与评估数据集来自相同分布的数据，即同一数据集中的数据。

提升经过充分预训练的语言模型在零样本设置下的表示，一种方法是使用中间数据 [33] 对其进行微调。这类中间数据包含由查询、相关文档和负例文档组成的三元组，也可以包含多个数据集。在使用语言模型的表示之前增加这一步骤，其优点在于能够缩小

[8]例如，在 TREC-DL-2020 段落和文档检索任务中，查询平均包含 5–6 个词，而段落和文档分别超过 50 个和 1000 个词；相比之下，在此处使用的信息寻求对话数据集中，对话上下文（查询）的平均长度因数据集而异，介于 70 至 474 个词之间，而响应（文档）则介于 11 至 71 个词之间。[9] 例如，可查看 MSMarco 基准排行榜上效果最佳的模型：https://microsoft.github.io/msmarco/。

预训练与当前下游任务之间的差距 [31,26,30]。在表2中，我们阐明了预训练、中间训练和微调之间的关系。

表2. 不同的训练阶段和数据、各自的用途、数据集示例，以及每个阶段结束后得到的稠密模型类型。

预训练数据	中间数据	目标数据
用途：学习通用表征表示	学习用于排序的句子表示	学习目标分布的表示
模型为零样本	零样本	微调
示例 Wikipedia	MSMarco	MANtIS

中间训练步骤通过对 Transformer 最后一层的输出应用平均池化函数，学习文本片段（查询和文档）的表示，随后利用这些表示计算点积相似度。损失函数使用同一批次中的多个负例文本，以对比方式学习表示，这也称为批内负采样。与直接使用 BERT 的$[CLS]$ 词元表示的朴素方法相比，这种做法能学到更好的文本表示 [33,2]。

函数 $f(\mathcal{U}, r)$ 随后表示为 $dot(\eta(concat(\mathcal{U})), \eta(r))$，其中 η 是对 Transformer 模型最后一层应用平均池化函数后得到的表示，而 $concat(\mathcal{U}) = u^1 \mid [U] \mid u^2 \mid [T] \mid ... \mid u^\tau$，其中 $\mid$表示拼接操作。上下文 $\mathcal{U}$ 中的话语与特殊分隔符 $[U]$ 和 $[T]$ 拼接，这些分隔符分别表示话语和轮次的结束[10]。零样本双编码器模型在对话领域中的有效性，是我们验证的第三项发现。

F4：微调密集检索 标准做法是使用来自模型评估所用同一数据集的目标数据对稠密模型进行微调。由于我们没有带标签的负响应，因此可以将数据集中的所有其他响应视为与对话上下文无关。计算正确响应相对于数据集中所有其他响应的概率可得 $P(r \mid \mathcal{U}) = {}^{P}{}_{(\mathcal{U},r)\sum_k P(\mathcal{U},r_k)}$。这种计算的成本高得难以承受，因此标准做法是使用少量负样本进行近似。负采~~样任务如~~下：给定对话上下文 $\mathcal{U}$，找出具有挑战性的响应 r^-，这些响应与 $\mathcal{U}$无关。负采样可以视为一种检索任务：使用 $\mathcal{U}$ 作为查询，对响应集合应用检索函数，从而利用模型检索负样本。

有了这样一个数据集，我们在中间步骤之后继续训练，方式与中间训练步骤相同，并使用以下交叉熵损失函数[11] 处理大小为 B 的批次：

[10] 在零样本设置下，特殊标记 $[U]$ 和 $[T]$ 不会获得任何有意义的表示，但可以在微调阶段学习这些表示。[11] 我们将此损失称为多负样本排序损失。

$$\mathcal{J}(\mathcal{U}, \mathbf{r}, \theta) = -\frac{1}{B}\sum_{i=1}^{B}\left[f\left(\mathcal{U}_i, r_i\right) - \log\sum_{j=1,j!=i}^{B} e^{f(\mathcal{U}_i, r_j)}\right],$$

其中， $f(\mathcal{U}, r)$ 是 Transformer模型最后一层均值池化结果的点积。微调双编码器模型在对话领域中的有效性，是我们在此验证的第四项发现。

F5：困难负采样 随机样本的一个局限是，排序模型可能很容易就能将它们与相关样本区分开来；而对于较难的负文档，模型仍可能难以应对。因此，另一种常见方法是使用排序模型，借助 BM25 等经典检索技术，根据给定查询检索负文档。这样找到的负文档在稀疏表示空间中更接近查询，因此是更难的负例。由于在许多有可用训练数据的情况下，稠密检索模型的表现优于稀疏检索，因此也有人提出了利用稠密检索的、更复杂的负采样技术 [46,12]。本文验证的第五项发现，是困难负采样对对话领域中的双编码器模型是否有效。

4 实验设置

为了比较不同的稀疏和稠密方法，我们考虑了三个大规模信息寻求对话数据集[12]：MSDialog [32] 包含 246K 个上下文-响应对，来源于微软 Answer社区这个微软多款产品的问答论坛中的 35.5K 段信息寻求对话；MANtIS [27] 包含 130 万个上下文-响应对，来自 Stack Exchange 的 14 个站点上的对话，例如 askubuntu 和 travel；$\text{UDC}_{\text{DSTC8}}$ [16]包含 184k 个从 Ubuntu IRC对话中拆分出的上下文-响应对。

实现细节 对于 BM25 和 BM25+RM3[13]，我们采用 Pyserini 的实现 [20]。为了训练 resp2ctxt扩展方法，我们使用 Huggingface transformers库 [44], 中的 t5-base 模型。我们对 T5 模型进行微调，训练 2 个周期，学习率为 2e-5，权重衰减为 0.01，批量大小为 5。使用 resp2ctxt 增强响应时，我们遵循 docT5query [25] 的方法，附加三种不同的上下文预测，并通过采样保留概率最高的前 10 个词汇标记。

对于零样本稠密模型，我们采用 SentenceTransformers [33]发布的模型。该库使用 Hugginface的transformers 来加载 BERT [4] 和 MPNet [37]等预训练模型。对于双编码器模型，我们使用

[12]MSDialog 可从 https://ciir.cs.umass.edu/downloads/msdialog/ 获取；MANtIS 可从 https://guzpenha.github.io/MANtIS/ 获取；$\text{UDC}_{\text{DSTC8}}$ 可从 https://github.com/dstc8-track2/NOESIS-II. 获取。[13] 我们通过网格搜索对扩展词数量、扩展文档数量和权重进行超参数调优。

预训练的 all-mpnet-base-v2 权重；在初步实验中，与其他预训练模型[14]相比，这些权重效果最佳。微调稠密检索模型时，我们采用多负样本排序损失。该损失函数接受多个困难负例，并利用批次中剩余的随机负例训练模型。对于对话上下文，我们总共使用 10 个负样本。我们对稠密模型进行微调，共训练 10k 步；每隔 100 步，就在一个重排序任务上评估模型，该任务要求从 10 个响应中选出相关响应。我们使用重排序验证集上的平均准确率均值，从整个训练过程中选出最佳模型用于评估。批量大小设为 5，预热步数占训练总步数的 10%。学习率为 2e-5，权重衰减为 0.01。我们使用 FAISS [13] 进行相似度搜索。

评估 为了评估检索系统的有效性，我们使用 $R@K$。因此，我们评估模型从所有可能的响应集合中找出正确响应的能力[15]。我们在 0.95 置信水平下进行学生 t 检验，并采用 Bonferroni 校正来比较不同方法在统计学上的显著性。比较基于每个对话上下文的结果进行。

5 结果

本节将依次讨论我们的实证结果，以及前人工作（第1节）提出的五项主要发现。表3列出了关于 F1至F4 的主要结果。表5列出了关于 F5 的结果。

F1 ✗ 通过 RM3 进行查询扩展优于不使用查询扩展 **[1,18,49,21]**。 BM25+ RM3（第 1b 行）在所分析的三个对话数据集上，均未优于 BM25（第 1a 行）。我们进行了全面的超参数微调，没有任何一组 RM3 超参数优于 BM25。 这表明，F1 不适用于对话响应检索任务。

论文的一位作者对从 60 个对话上下文样本中新增的词语进行了人工分析，结果显示，根据我们最审慎的判断，只有 18% 的样本至少新增了一个相关词语。网络搜索中的查询往往不完整、描述不充分且含义模糊；与之不同的是，在此处采用的信息寻求数据集中，对话上下文（查询）相当详尽，且词语比响应（文档）更多。我们推测，由于对话上下文本身已经相当详尽，因此在这一领域中，扩展任务更为棘手，这也解释了为何我们观察到许多对话新增的词语都包含噪声。

[14] 我们考虑的替代模型列于 https://www.sbert.net/docs/pretrained_models.html 的模型概览部分。[15] 对话响应排序 [50,8,39] 的标准评估指标是位置 K 上的召回率，其中候选项数量为 n $R_n@K$。由于我们关注的是第一阶段检索，因此将 n 设为整个答案集合

表3。F1–F4 泛化能力的结果。粗体数值表示每种方法中最高的召回率。上标表示使用学生 t 检验并进行 Bonferroni 校正后，结果具有统计显著性，*1=other methods from the same group1=best from unsupervised sparse retrieval; 2=best from supervised sparse retrieval; 3=best from zero-shot dense retrieval.* $3^{\dagger}$ 例如，在 F 中表示第 (3d) 行优于第 (3a–c) 行，^{1} 表示其优于第 (1a) 行，^{2} 表示其优于第 (2b) 行。

	MANtIS		MSDialog		UDC_{DSTC8}	
	R@1	R@10	R@1	R@10	R@1	R@10
(0) 随机	0.000	0.000	0.000	0.001	0.000	0.001
无监督式稀疏检索				**F1**		
(1a) BM25	**0.133**†	**0.299**†	**0.064**†	**0.177**†	**0.027**†	**0.070**†
(1b) BM25 + RM3	0.073	0.206	0.035	0.127	0.011	0.049
监督式稀疏检索				**F2**		
(2a) BM25 +resp2ctxt	0.135	0.309	0.074	**0.208**	0.028	0.067
(2b) BM25 + $resp2ctxt_{lu}$	**0.147**$^{\dagger 1}$	**0.325**$^{\dagger 1}$	**0.075**^{1}	0.202^{1}	**0.029**	**0.076**
零样本稠密检索 ($Model_{IntermediateData}$)				**F3**		
(3a) $ANCE_{600K-MSMarco}$	0.048	0.111	0.050	0.124	0.010	0.028
(3b) $TAS\text{-}B_{400K-MSMarco}$	0.062	0.143	0.060	0.157	0.019	0.050
(3c) 双编码器$_{215M-mul}$	0.138	0.297	0.108	0.277	0.023	0.076
(3d) 双编码器$_{1.17B-mul}$	**0.155**$^{\dagger 1}$	**0.341**$^{\dagger 12}$	**0.147**$^{\dagger 12}$	**0.339**$^{\dagger 12}$	**0.041**†	**0.097**$^{\dagger 12}$
微调密集检索器 ($Model^{NegativeSampler}$)				**F4**		
(4a) 双编码器$^{Random(0)}$	**0.130**	**0.307**	**0.168**^{123}	**0.387**^{123}	**0.050**^{12}	**0.128**^{123}

F2 ✓ 通过 resp2ctxt 进行文档扩展，相较于不进行扩展有所改进 [**25,21,19**]。我们发现，在三个数据集中的两个数据集上，采用朴素的响应扩展方法略有改善，BM25+resp2ctxt (2a) 的表现优于 BM25 (1a)。然而，正如 BM_{lu} resp2ctxt25+$resp2ctxt_{lu}$ 的较高召回率所示，所提出的改进方法仅预测对话的最后一句话（2b），表现优于预测整句话。对于 MANtIS数据集，使用经过训练以预测对话上下文的模型时，R@10 为 0.309；使用经过训练以仅预测对话上下文最后一句话的模型时，R@10 则升至 0.325。 因此，我们发现 F2 能够推广到对话响应检索，尤其是在仅预测上下文的最后一句话时[16]。

为了了解响应扩展方法主要在做什么——术语重新加权还是添加新词项，我们在表4中列出了两种方法所添加的新词项百分比。表中显示，与 resp2ctxt 相比，$resp2ctxt_{lu}$进行术语重新加权的程度高于添加新词

[16] 未来的工作可以采用更复杂的技术，以确定应预测对话上下文中的哪些部分。

表4。resp2ctxt 和 resp2ctxt$_{lu}$ 的增强统计数据。新词是指此前不存在于文档中的词语。

	MANtIS	MSDialog	UDC$_{DSTC8}$
上下文平均长度	474.12	426.08	76.95
响应平均长度	42.58	71.38	11.06
增强平均长度 - resp2ctxt	494.23	596.99	202.3
增强 平均长度 - resp2ctxt$_{lu}$	138.5	135.29	72.57
新词百分比 - resp2ctxt	71%	69%	71%
新词百分比 - resp2ctxt$_{lu}$	59%	37%	63%

（两者平均分别有 53% 和 70% 是新词，因此，分别有 47% 和 30% 是通过添加已有词语来改变权重），生成的总体扩展规模较小（平均分别为 115.45 和 431.17）。

F3 ✓ 稀疏检索优于零样本稠密检索 [**34,41**]。如第 (1a–b) 行和第 (2a–b) 行与第 (3a–c) 行的比较所示，稀疏检索模型比大多数零样本稠密模型更有效。不过，正如表3中第 (3d) 行与第 (2b) 行的比较所示，经过包含对话在内的大规模、多样化数据集上的中间训练后，稠密检索模型比强大的稀疏检索模型更有效。

例如，对于 MSDialog 数据集，单纯基于 MSMarco 数据集的零样本稠密检索模型 (3a–b) 在 R@10 指标上的表现平均比强大的稀疏基线 (2b) 差 35%；而使用多样化数据训练、包含 1.17B 个实例的零样本模型 (3d) 则比稀疏基线 (2b) 好 68%。使用更多中间训练数据[17], 时，我们发现，零样本稠密检索模型 (3d) 能够在各数据集上平均以 33% 的 R@10 优势超过稀疏检索基线。

因此，我们表明，只有在不使用大量多样化中间数据的情况下，F3 才能泛化到对话响应检索。正如预期的那样，中间训练数据分布越接近评估数据，稠密检索模型的表现就越好。结果表明，优秀的零样本检索模型需要在来自多个数据集的大规模训练数据上进行中间训练，才能很好地泛化到不同领域，并超越强大的稀疏检索基线。

F4 ✓ 可访问目标训练数据的稠密模型优于稀疏模型 [**7,15,34**]。 首先，我们看到，经过中间训练的稠密检索模型采用随机采样进行微调——表3中的第 (4a) 行——在三个数据集中的两个上取得了总体最佳效果。 这一结果表明，F4 可以泛化到

[17]有关中间数据的完整说明，请参阅 https://huggingface.co/sentence-transformers/all-mpnet-base-v2。

在采用中间训练时的对话响应检索任务[18]。与只能访问中间训练数据相比，能够访问目标数据意味着模型学到的表示更接近数据的真实分布。

我们假设，对 MANtIS（4a）的双编码器进行微调会产生负面影响，因为中间数据包含 Stack Exchange 的响应。因此，MANtIS 涵盖的 Stack Exchange 对话集可能只会导致中间表示过拟合。作为这一假设的证据，我们发现：（I）学习曲线很快趋于平稳（不同于其他数据集）；（II）对微调过程中未使用 Stack Exchange 数据的另一个语言模型（MSMarco）进行微调后，双编码器$_{bert-base}$（3c）的效果显著提升，R@10 从 0.092 提高到 0.205。

F5 ✓ 对于训练稠密检索模型，困难负采样优于随机采样 [**46,51**]。令人意外的是，我们发现，直接使用效果更好的模型来选择负候选项，会降低稠密检索模型的效果（见表5中的困难负采样）。在使用不同语言模型、对所有数据集启用或关闭中间训练，以及使用不采用批内负采样的替代对比损失时，我们都观察到了这一现象 [10] [19]。

在检验了多种可能解释“更难的负例为何无法提升稠密检索模型效果”的假设后，我们发现，采用更好的负采样方法时，假负例样本会显著增加。假负例是指对于上下文而言可能有效的响应。这些相关响应被标记为负例，因而会导致模型遗忘上下文与响应之间的相关匹配。下方是双编码器模型检索到的一个假负例样本示例（表3中的第3d行）：

对话上下文（$\mathcal{U}$）： 嘿.....Dapper 还要多久才会发布？ [U] 14 几天 [.....] [U] 我以为今晚就会发布 正确响应（r^+）： 开个玩笑，大概还要几个小时 假负例样本（r^-）： Dapper 有可能会延期 [.....] 与此同时，Dapper 的讨论应该在 Ubuntu 中进行+1

去噪技术尝试通过减少假阴性的数量来解决这个问题。我们采用了一种简单的方法：不使用排名靠前的响应作为负响应，而是使用排名靠前的响应中的末尾响应作为负样本[20]。这样可以降低获得假阳性的可能性；如果 $k << |\mathcal{D}|$，我们就不会获得随机样本。表5中的实验结果表明，这种去噪技术（第(3b)行）提高了较难负样本上的效果，在三个数据集中的两个上胜过了表3中的所有模型

[18] 实验结果表明，如果不采用中间训练步骤，微调密集检索器的泛化能力较差；对于 MANtIS、MSDialog 和 UDC，行 (3d) 的 R@10 性能分别降至 0.172、0.308 和 0.063。DSTC8 由于篇幅限制，此处未展示结果。[19] 例如，如果我们检索 $k = 100$ 个响应，而不是使用排名前 1–10 位的响应，就改为使用列表末尾排名第 91–100 位的响应。

。结果表明，只有采用去噪技术时，F5 才能泛化到对话响应检索任务。

表5. F5 泛化能力的结果——采用和未采用困难负采样去噪策略。上标表示通过带 Bonferroni 校正的学生 t 检验得出的统计显著提升。†=*significance against the random sampling baseline* ‡=*significance against hard negative sampling without denoising.* ,

	MANtIS	MSDialog	$\text{UDC}_{\text{DSTC8}}$
	R@10	R@10	**R@10**
基线			
(1) 双编码器$_{Random}$	0.307	0.387	**0.128**
困难负采样			
(2a) 双编码器$_{BM25}$	0.271	0.316	0.087
(2b) 双编码器$_{Bi-encoder}$	0.146	0.306	0.051
去噪困难负采样			
(3a) 双编码器$_{BM25}$	0.257	0.358‡	0.121‡
(3b) 双编码器$_{Bi-encoder}$	**0.316**†‡	**0.397**†‡	0.107‡

6 结论

在这项工作中，我们检验了从段落和文档检索任务的实验中获得的稠密检索和稀疏检索知识，是否能够推广到对话响应的第一阶段检索。我们的可复现性研究表明，虽然大多数发现都能推广到我们的领域，但简单地迁移模型并不总是奏效。仔细分析相关领域，或许能发现更好的技术适配方法。

作为未来工作，我们认为一个重要方向是评估能够对查询和文档进行加权与扩展的学习式稀疏方法 [5]——虽然 resp2ctxt 既能通过重复现有词项来调整响应中词项的权重，也能通过添加新词项来扩展词项，但它无法对对话上下文进行加权与扩展。

致谢 本研究得到了荷兰科学研究组织项目 SearchX (639.022.722) 和 NWO Aspasia (015.013.027) 的支持。

参考文献

1. Abdul-Jaleel, N., Allan, J., Croft, W.B., Diaz, F., Larkey, L., Li, X., Smucker, M.D., Wade, C.：Umass 参加 TREC 2004：新颖性与难例。计算机科学系教职工出版物系列，第 189 页（2004）

2. Aghajanyan, A., Gupta, A., Shrivastava, A., Chen, X., Zettlemoyer, L., Gupta, S.：Muppet：通过预微调实现大规模多任务表示。arXiv 预印本 arXiv:2101.11038（2021） 3. Anand, A., Cavedon, L., Joho, H., Sanderson, M., Stein, B.：对话式搜索（Dagstuhl 研讨会 19461）。载于：Dagstuhl 报告。第 9 卷。Schloss Dagstuhl-Leibniz- Zentrum f¨ur Informatik（2020） 4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.：Bert：用于语言理解的深度双向 Transformer 预训练。arXiv 预印本 arXiv:1810.04805（2018） 5. Formal, T., Lassance, C., Piwowarski, B., Clinchant, S.：SPLADE v2：用于信息检索的稀疏词汇与扩展模型。arXiv 预印本 arXiv:2109.10086（2021） 6. Furnas, G.W., Landauer, T.K., Gomez, L.M., Dumais, S.T.：人机通信中的词汇问题。ACM 通讯 30(11)，964–971（1987） 7. Gao, L., Callan, J.：面向稠密段落检索的无监督语料库感知语言模型预训练。arXiv 预印本 arXiv:2108.05540（2021） 8. Gu, J.C., Li, T., Liu, Q., Ling, Z.H., Su, Z., Wei, S., Zhu, X.：用于基于检索的聊天机器人多轮响应选择的说话人感知 BERT。载于：第 29 届 ACM 信息与知识管理国际会议论文集。第 2041–2044 页（2020） 9. Gu, J.C., Ling, Z.H., Liu, Q.：用于基于检索的聊天机器人多轮响应选择的交互式匹配网络。载于：第 28 届 ACM 信息与知识管理国际会议论文集。第 2321–2324 页（2019） 10. Hadsell, R., Chopra, S., LeCun, Y.：通过学习不变映射实现降维。载于：2006 年 IEEE 计算机学会计算机视觉与模式识别会议（CVPR'06）。第 2 卷，第 1735–1742 页。IEEE（2006） 11. Han, J., Hong, T., Kim, B., Ko, Y., Seo, J.：用于改进基于检索的对话系统的细粒度后训练。载于：2021 年北美计算语言学协会分会会议：人类语言技术论文集。第 1549–1558 页。计算语言学协会，线上（2021 年 6 月）。https://doi.org/10.18653/v1/2021.naacl-main.122，https://aclanthology.org/2021.naacl-main.122 12. ¨Lin, S.C., Yang, J.H., Lin, J., Hanbury, A.：通过均衡的主题感知采样，高效地训练高效的稠密检索器。载于：第 44 届国际 ACM SIGIR 信息检索研究与发展会议论文集。第 113–122 页（2021） 13. Johnson, J., Douze, M., J´egou, H.：利用图形处理器进行十亿规模的相似度搜索。IEEE 大数据汇刊 7(3)，535–547（2019） 14. Kadlec, R., Schmid, M., Kleindienst, J.：改进的 Ubuntu 语料库对话深度学习基线。arXiv 预印本 arXiv:1510.03753（2015） 15. Karpukhin, V., O˘guz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., Yih, W.t.：用于开放域问答的稠密段落检索。arXiv 预印本 arXiv:2004.04906（2020） 16. Kummerfeld, J.K., Gouravajhala, S.R., Peper, J.J., Athreya, V., Gu- nasekara, C., Ganhotra, J., Patel, S.S., Polymenakos, L.C., Lasecki, W.：用于对话解缠的大规模语料库。计算语言学协会第 57 届年会论文集（2019）。https://doi.org/10.18653/v1/p19-1374，http://dx.doi.org/10.18653/v1/P19-1374

17. Lan, T., Cai, D., Wang, Y., Su, Y., Mao, X.L., Huang, H.：探索用于对话回复选择的稠密检索。arXiv 预印本 arXiv:2110.06612（2021）18. Lin, J.：尽可能简单且可行的方法：使用文本分类进行伪相关反馈。arXiv 预印本 arXiv:1904.08861（2019）19. Lin, J.：针对信息检索表征方法提出的概念框架。arXiv 预印本 arXiv:2110.01529（2021）20. Lin, J., Nogueira, R., Lin, S.C., Yang, J.H., Pradeep, R., Nogueira, R.：Pyserini：一个用于基于稀疏和稠密表示开展可复现信息检索研究的 Python 工具包。载于：第44届年度国际 ACM SIGIR 信息检索研究与发展会议（SIGIR 2021）论文集。第2356–2362页（2021）21. Lin, J., Nogueira, R., Yates, A.：文本排序的预训练 Transformer：从 BERT 到更多。人类语言技术综合讲座 14(4)，1–325（2021）22. Lin, Z., Cai, D., Wang, Y., Liu, X., Zheng, H.T., Shi, S.：世界并非非黑即白：利用灰度数据学习对话回复选择排序。arXiv 预印本 arXiv:2004.02421（2020）23. Lowe, R., Pow, N., Serban, I., Pineau, J.：Ubuntu对话语料库：面向非结构化多轮对话系统研究的大型数据集。arXiv 预印本 arXiv:1506.08909（2015）24. Nogueira, R., Cho, K.：使用 BERT 进行段落重排序。arXiv 预印本 arXiv:1901.04085（2019）25. Nogueira, R., Lin, J., Epistemic, A.：从 doc2query 到 doctttttquery。线上预印本 6（2019）26. Peeters, R., Bizer, C., Glavaˇs, G.：用于产品匹配的 BERT 中间训练。small 745(722), 2–112（2020）27. Penha, G., Balan, A., Hauff, C.：介绍 MANtIS：一种新型多领域信息寻求对话数据集。arXiv 预印本 arXiv:1912.04639（2019）28. Penha, G., Hauff, C.：信息检索的课程学习策略：对话响应排序的实证研究。arXiv 预印本 arXiv:1912.08555（2019）29. Penha, G., Hauff, C.：对话式搜索系统评估中的挑战。载于：Converse@ KDD（2020）30. Poth, C., Pfeiffer, J., R¨uckl´e, A., Gurevych, I.：应该对什么进行预训练？高效的中间任务选择。arXiv 预印本 arXiv:2104.08247（2021）31. Pruksachatkun, Y., Phang, J., Liu, H., Htut, P.M., Zhang, X., Pang, R.Y., Vania, C., Kann, K., Bowman, S.R.：使用预训练模型进行自然语言理解的中间任务迁移学习：何时有效？为何有效？arXiv 预印本 arXiv:2005.00628（2020）32. Qu, C., Yang, L., Croft, W.B., Trippas, J.R., Zhang, Y., Qiu, M.：分析并描述信息寻求对话中的用户意图。载于：第41届 ACM 信息检索研究与发展国际会议。第989–992页（2018）33. Reimers, N., Gurevych, I.：Sentence-BERT：使用孪生 BERT 网络生成句子嵌入。载于：2019年自然语言处理经验方法会议论文集。计算语言学协会（2019年11月），https://arxiv.org/abs/1908 10084 34 . . Ren, R., Qu, Y., Liu, J., Zhao, W.X., Wu, Q., Ding, Y., Wu, H., Wang, H., Wen, J.R.：对零样本稠密检索进行全面考察。arXiv 预印本 arXiv:2204.12755 （ 2022 ） 35. Robertson, S.E., Walker, S.：对概率加权检索二泊松模型的一些简单有效近似。载于：SIGIR'94。第232–241页。Springer（1994）

36. Robinson, J., Chuang, C.Y., Sra, S., Jegelka, S.：使用困难负样本进行对比学习。arXiv 预印本 arXiv:2010.04592（2020）37. Song, K., Tan, X., Qin, T., Lu, J., Liu, T.Y.：MPNet：用于语言理解的掩码与置换预训练。神经信息处理系统进展 33，16857–16867（2020）38. Tao, C., Feng, J., Liu, C., Li, J., Geng, X., Jiang, D.：通过互学习构建高效且有效的基于检索的对话系统。arXiv 预印本 arXiv:2110.00159（2021）39. Tao, C., Wu, W., Xu, C., Hu, W., Zhao, D., Yan, R.：面向基于检索的聊天机器人的多轮响应选择的多表示融合网络。载于：WSDM。第267–275页（2019）40. Thakur, N., Reimers, N., Daxenberger, J., Gurevych, I.：增强型 SBERT：一种用于改进双编码器成对句子评分任务的数据增强方法。arXiv 预印本 arXiv:2010.08240（2020）41. Thakur, N., Reimers, N., R¨uckl´e, A., Srivastava, A., Gurevych, I.：BEIR：用于信息检索模型零样本评估的异构基准。arXiv 预印本 arXiv:2104.08663（2021）42. Whang, T., Lee, D., Lee, C., Yang, K., Oh, D., Lim, H.：一种有效的领域自适应后训练方法，用于响应选择中的 BERT。arXiv 预印本 arXiv:1908.04812（2019）43. Whang, T., Lee, D., Oh, D., Lee, C., Han, K., Lee, D.h., Lee, S.：响应选择模型真的知道接下来会发生什么吗？面向多轮响应选择的话语操纵策略。载于：AAAI 人工智能大会论文集。第35卷，第14041–14049页（2021）44. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., 等：Hugging Face 的 Transformers：先进的自然语言处理。arXiv 预印本 arXiv:1910.03771（2019）45. Wu, Y., Wu, W., Xing, C., Zhou, M., Li, Z.：序列匹配网络：一种用于基于检索的聊天机器人的多轮响应选择的新架构。载于：ACL。第496–505页（2017）46. Xiong, L., Xiong, C., Li, Y., Tang, K.F., Liu, J., Bennett, P., Ahmed, J., Over- wijk, A.：用于稠密文本检索的近似最近邻负对比学习。arXiv 预印本 arXiv:2007.00808（2020）47. Yang, L., Qiu, M., Qu, C., Chen, C., Guo, J., Zhang, Y., Croft, W.B., Chen, H.：IART：信息寻求型对话系统中基于 Transformer 的意图感知响应排序。arXiv 预印本 arXiv:2002.00571（2020）48. Yang, L., Qiu, M., Qu, C., Guo, J., Zhang, Y., Croft, W.B., Huang, J., Chen, H.：信息寻求型对话系统中结合深度匹配网络与外部知识的响应排序。载于：SIGIR。第245–254页（2018）49. Yang, W., Lu, K., Yang, P., Lin, J.：批判性审视“神经网络热潮”：神经排序模型的弱基线与效果提升的可加性。载于：第42届 ACM 信息检索研究与发展国际会议论文集。第1129–1132页（2019）50. Yuan, C., Zhou, W., Li, M., Lv, S., Zhu, F., Han, J., Hu, S.：用于基于检索的聊天机器人多轮响应选择的多跳选择网络。载于：EMNLP。第111–120页（2019）51. Zhan, J., Mao, J., Liu, Y., Guo, J., Zhang, M., Ma, S.：利用困难负例优化稠密检索模型训练。载于：第44届国际 ACM SIGIR 信息检索研究与发展会议论文集。第1503–1512页（2021）

52. Zhang, Z., Zhao, H.：多轮对话理解进展：综述。arXiv 预印本 arXiv:2110.04984（2021）53. Zhang, Z., Zhao, H.：面向对话理解的结构化预训练。arXiv 预印本 arXiv:2105.10956（2021）54. Zhou, X., Li, L., Dong, D., Liu, Y., Chen, Y., Zhao, W.X., Yu, D., Wu, H.：采用深度注意力匹配网络的聊天机器人多轮响应选择。载于：第56届计算语言学协会年会论文集（第1卷：长篇论文）。第1118–1127页（2018）