**Augmentation-Adapted Retriever Improves Generalization of Language Models as Generic Plug-In**

## 太长不看版

这篇论文探讨了一种通用的检索插件方案，名为增强适应检索器（Augmentation-Adapted Retriever），旨在改善语言模型（LMs）的泛化能力。在知识密集型任务中，检索增强可以通过为LMs提供外部信息来帮助它们。AAR的设计目标是作为通用插件，为不可见的目标LMs提供有用的文档。AAR从已知源LM中学习LM的偏好，以检索适用于目标LM的文档。

# 摘要

检索增强（retrieval augmentation）可以通过向语言模型(LMs)提供外部信息来帮助它们完成知识密集型任务。以往的检索增强工作通常是联合对检索器和LM进行微调，使它们紧密耦合。在本文中，作者使用通用检索插件的方案:检索器是用来辅助可能事先不知道或无法一起微调的目标lm。为了从未知的目标LM中检索有用的文档，作者提出了**增强适应检索器-AAR（augmentation-adapted retriever）**，**它从已知的源LM中学习LM的偏好**。在MMLU和PopQA数据集上的实验表明，用小源LM训练的AAR能够显著提高从250M Flan-T5到175B InstructGPT的更大的目标LM的zero-shot泛化。进一步的分析表明，不同LM的偏好是重叠的，这使得使用单个源LM训练的AAR可以作为各种目标LM的通用插件。

作者源码开源地址：[OpenMatch/Augmentation-Adapted-Retriever: [ACL 2023] This is the code repo for our ACL'23 paper "Augmentation-Adapted Retriever Improves Generalization of Language Models as Generic Plug-In". (github.com)](https://github.com/OpenMatch/Augmentation-Adapted-Retriever)

# 引言

拥有数十亿个参数的大型语言模型(LMs)能够捕获大量的人类知识，从而在各种下游任务上取得一致的改进。然而，大型LMs不可否认的缺点在于其高计算成本，这对其效率产生了负面影响。此外，从预训练和隐式推理过程中记忆的知识有时会不准确和难以处理，从而阻碍了它们在知识密集型任务中的应用。

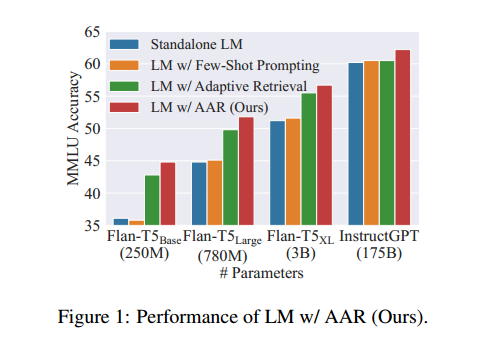
与利用LMs参数中嵌入的知识和推理能力不同，检索增强用一个可以从外部语料库检索知识的检索器来增强LM。另一方面，先前的检索增强方法需要对主干网络LM进行微调，以适应检索器并处理特定的下游任务。当越来越多的独特需求出现时，这种微调可能会很昂贵。更重要的是，许多顶级LMs只能通过黑盒APIs访问。这些APIs允许用户提交查询和接收响应，但通常不支持微调。

作者引入了Augmentation Adapted Retriever (AAR)作为通用插件来帮助黑盒LMs处理下游任务。为了向许多不可见的LMs检索有价值的文档，作者建议利用一个小源LM来为检索器的训练提供LM偏好信号。训练后的检索器(即AAR)可以直接通过插入检索到的文档来辅助大型目标LM。

选择一个小型encoder-decoder LM作为源LM，并利用fusion-in-decoder attention分数标注LM偏好文档。然后将LM偏好的文档与人类偏好的文档结合起来，形成正面文档集。负面文档由检索器自己使用ANCE 技术进行挖掘。通过对LM的偏好进行微调后，可以直接辅助未知目标LM进行零射击任务泛化。

在多任务语言理解数据集MMLU 和以实体为中心的问答数据集PopQA上评估AAR。对于目标LMs，作者选择Flan-T5系列作为encoder-decoder LMs的主干网络，选择InstructGPT系列作为 decoder-only LMs的主干网络。图1显示，在通用AAR的帮助下，不同大小和体系结构的LMs可以始终优于独立LMs;较小的lm的性能有时会明显超过较大尺寸的独立lm(例如，Flan-T5Large w/ AAR比独立的Flan-T5XL高出0.6%)。

AAR也显示出优于其他增强方法的优势，如few-shot提示和自适应检索.



# 相关工作

**检索增强**。使用从外部记忆中检索到的信息来增强LMs在各种知识密集型任务中显示出有效的效果。先前的工作探索了以端到端方式训练整个检索器-LM系统的新方法，使用检索增强序列对数似然，fusion-in-decoder attention**蒸馏**或知识图谱。在这项工作中，作者设计了一个检索器，它可以作为一个通用插件来帮助各种看不见的LMs。

**zero-shot学习和推理**。GPT3、GPT-4和PaLM等大规模无监督预训练的LMs能够在许多下游任务上执行zero-shot学习，并在推理时提供任务描述。指令微调lm ，使用人类指令对多个监督任务进行预训练，也表现出强大的zero-shot学习能力。作者通过利用检索到的信息提高了LMs的zero-shot泛化。证明了识别LMs训练检索器的偏好可以反过来为LMs带来额外的证据文本。

# **方法**

## 初步概念

检索增强LM是一种利用外部信息来提高其表现的LM。它使用检索器从语料库中检索相关文档，然后利用这些文档来增强其语言生成能力。

检索器的目标是从语料库中找到一个增强文档集，以帮助LM处理给定的查询。先前研究主要关注在密集向量空间中搜索的密集检索系统，因为密集检索通常比稀疏检索更准确、更有效。

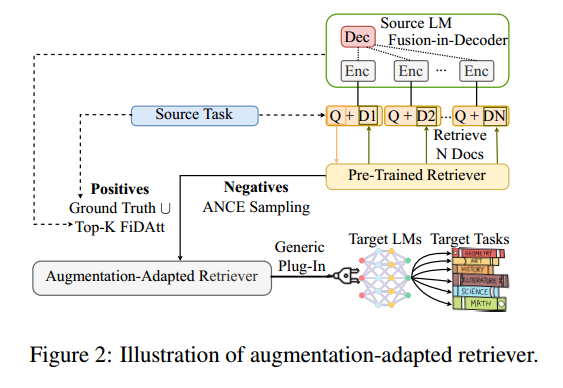
密集检索模型首先使用预训练的encoder 将和文档d表示为embedding空间，  
并通过点积函数f匹配它们的embedding，支持近似最近邻搜索(ANN) 。然后作者将包含top-N检索文档的定义为:  
  
  
 对于LM主干网络，decoder-only模型和encoder-decoder模型是检索增强LMs的两种主要选择。

给定一个像GPT-3 这样的decoder-only的LM, LM输入可以是查询和所有增强文档。然后，LM将根据输入自动回归生成答案。

对于像T5这样的encoder-decoder LM，采用简单的连接作为encoder输入可能仍然有效。但是，由于与文档数量相关的二次self-attention计算，该方法可能无法扩展到大量文档。为了更有效地聚合多个文档，Izacard和Grave 提出了fusion-in-decoder (FiD) attention机制，该机制很快成为encoder-decoder检索增强LMs发展的主流。它首先对对的每个连接分别进行编码，然后让decoder处理所有部分:  
  
这样，编码器一次计算一个文档的自关注，使得计算成本可以随着文档的数量线性增长。此外，从LM的角度，FiD cross-attention可以有效地估计增强文档的相对重要性方面。因此，软FiD蒸馏，最小化了检索似然和LM似然之间的KL-divergence，通常用于端到端训练检索器和LM。

## 增强适应检索器（Augmentation-adapted Retriever）

由于现实世界中不断出现的需求和黑盒APIs的限制，为每个可能的下游任务微调检索增强的LM可能是不可行的。因此，作者引入增强适应检索器(AAR)作为黑盒LMs的通用插件。如图2所示，AAR可以学习LMs的偏好，而不需要对它们进行微调。



利用encoder-decoder LM作为源LM ()，在源任务()上提供LM偏好信号，用于微调预训练的检索器。然后，作者将微调后的检索器插入与不相交的一组目标任务()上的不可见的LM ()中。

训练方法从一个源任务开始，作者将源LM 的平均FiD cross-attention(FiDAtt)分数与文档对应，从所有层的第一个decoder token，所有注意力头和所有输入token t (⊕q):

ln,hn,tn分别是层数，头部数和输入token数。

为了使训练过程更加稳健，作者利用FiDAtt分数以离散的方式标注LM偏好的积极文档:

其中是上人类偏好的正文档集(即图中ground truth)。表示在检索的文档集中top-k平均FiDAtt分数的文档。

然后，作者根据ANCE 对难负采样，并将检索器的训练损失表示为:  
  
  
其中*M*为负采样深度的超参数，*l*为标准交叉熵损失。在对检索器进行微调后，作者直接使用它来增强目标任务集中每个任务的不可见目标LM 。

# 四．实验方法

## 目标任务

以MMLU和PopQA为目标任务。

MMLU是一个多任务语言理解数据集，包含57个选择题回答子任务。这些子任务一般可以分为四类:人文科学、社会科学、STEM（科学（Science），技术（Technology），工程（Engineering），数学（Mathematics））和其他。作者对每个类别中的子任务的准确率取平均值，从而得到最终的分数。作者报告了实验在验证集上的准确性。

PopQA是一个以实体为中心的问答数据集，主要集中在长尾问题上。作者同样在主要实验中报告了验证集上的准确性。

## 作者方法

**检索器。**采用两种广泛使用的检索器来初始化AAR:从T5Base初始化ANCE ，从BERTBase初始化Contriever。它们之前都在MS MARCO 上进行了微调。对于检索语料库，作者选择MS MARCO作为MMLU，选择KILT-Wikipedia 作为PopQA。

**LM。**采用Flan-T5系列作为encoder-decoder LM的主干网络，采用InstructGPT1 系列作为decoder-only LM的主干网络。这些模型已经过多任务指令微调，并广泛用于评估zero-shot泛化。

**实现细节。**使用MS MARCO 作为源任务，因为它是训练检索器的常见选择。该数据集由高质量的问题组成，这些问题需要现实世界的知识来回答，这与目标任务非常一致，并且没有重叠。

考虑到实现效率，将Flan-T5Base作为源LM，将更大的模型作为目标LM。

在增强适应训练中，直接设置总文档数N = 10，偏好文档数K = 2，负挖掘深度M = 100。在单个A100 GPU (40G)上运行所有实验。

## 基线

**Zero-shot设置。**作者将其的方法与最先进的zero-shot基线进行比较。独立LMs，包括Flan-T5、InstructGPT、GAL和OPT-IML-Max，是由描述所需任务和问题的自然语言指令提示的。自适应检索根据问题的受欢迎程度，选择性地利用非参数存储器(检索增强)和参数存储器(从预训练中获得的知识)。

在作者的主要实验中，从他们的论文中选择了最佳组合，其中包括Contriever作为非参数存储器和GenRead作为参数存储器。

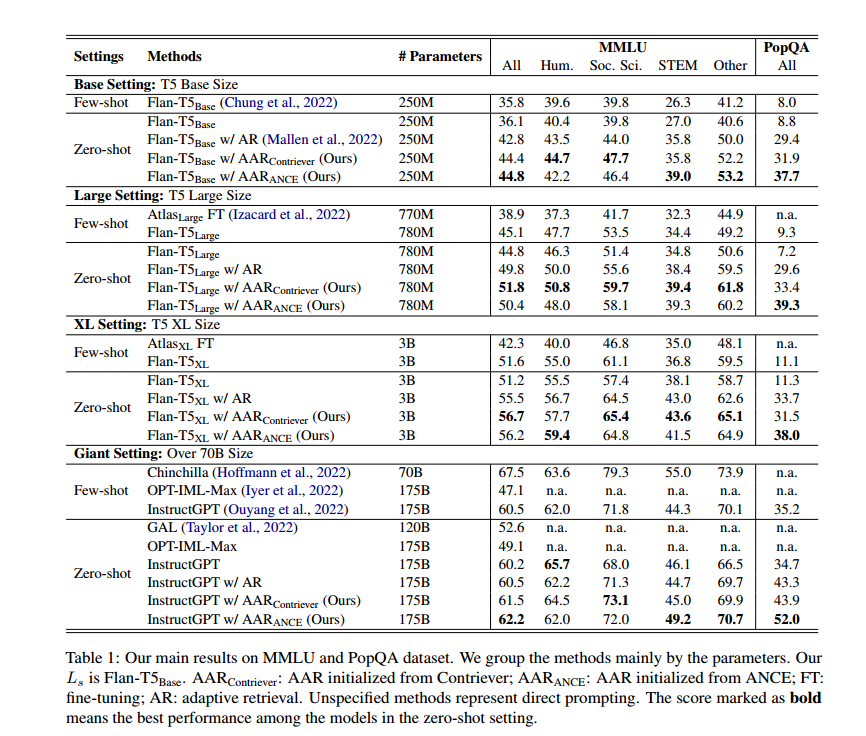
**Few-shot设置。**作者还纳入了以往的few-shot模型的结果，以供参考。Flan-T5、InstructGPT、Chinchilla 和OPT-IML-Max采用了few-shot展示，这为LMs提供了有限数量的任务示例。这使得模型能够从这些例子中进行泛化，并产生准确的响应。

Atlas是一种最先进的检索增强LM，它与LM联合利用无监督数据对检索器进行预训练，通过few-shot数据的attention蒸馏对检索器进行微调。

# 五．验证结果

## 整体性能

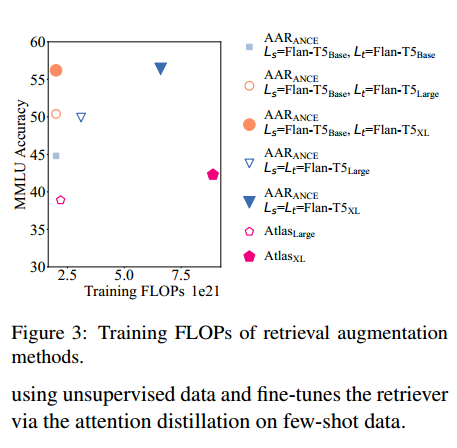
表1表明，在通用AAR的帮助下，不同大小和体系结构的目标LMs在zero-shot 设置中可以显著优于其独立基线。值得注意的是，AAR甚至将强大的InstructGPT在MMLU上提高了2%，在PopQA上提高了近20%。作者假设PopQA数据集主要由长尾问题组成，因此需要更多的增强信息来获得较高的准确性。AAR优于其他增强方法，如few-shot prompting和adaptive retrieval，因为它们可能不像AAR那样提供广泛的证据文本。



同时，AAR是一种高效的增强方法，因为它只依赖于一个小的源LM

Flan-T (250M)来提供训练信号，可以很好地泛化到更大容量的目标LM。

图3说明，仅仅将源LM设置为目标LM(用倒三角表示)并不能显著提高MMLU的准确性。然而，这可能会使所需的培训预算增加三倍。只有使用一个小的源LM才能以更少的训练FLOPs大大胜过强大的Atlas。

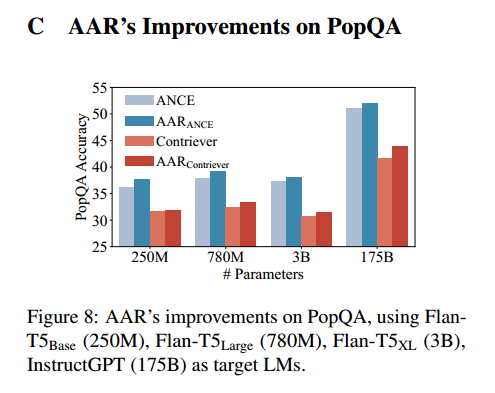
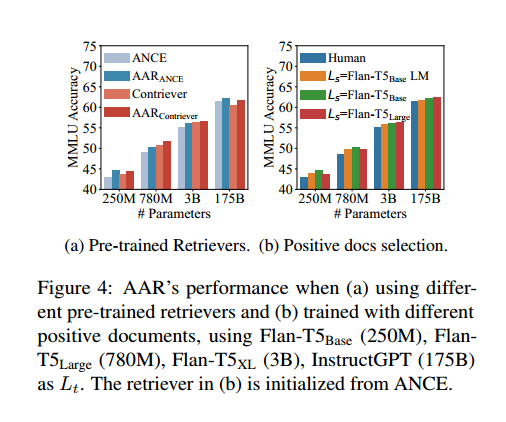


## 消融实验

在本实验中，作者进行了增强适应训练的消融研究，并分析了模型在训练过程中的行为。

图4a表明，与预先训练的检索器相比，增强适应训练可以带来额外的改进。一般来说，ANCE比Contriever从增强训练中获益更多。这可能是因为Contriever已经在海量数据增强和MS MARCO上进行了密集的预训练，而ANCE只在MS MARCO上进行了训练。在表7中提供了精确的数字，在图8中提供了PopQA结果，它们产生了与MMLU相似的观察结果。

在图4b中，作者比较了使用不同正面文档训练的检索器，包括由搜索用户注释的人类偏好文档(蓝色条)，由源LM获得的LM偏好文档(橙色条)，以及它们的组合(绿色条和红色条)。由于检索器已经在用户注释的MS MARCO上进行了预训练，简单地使用人类偏好的文档来训练它可能是没有意义的，因此在所有方法中表现最差。只使用LM偏好的文档比只使用人类偏好的文档有显著的好处，**合并人类偏好和LM偏好的文档(作者的主要设置)进一步提高了检索器的性能。**最后，当目标LM相对较小时，使用Flan-T5Base作为源LM比使用Flan-T5Large产生更好的结果。但是，随着目标LM的大小增加，这两种方法的性能相当。**所以选择在增强适应训练中使用小源LM是合理有效的。**



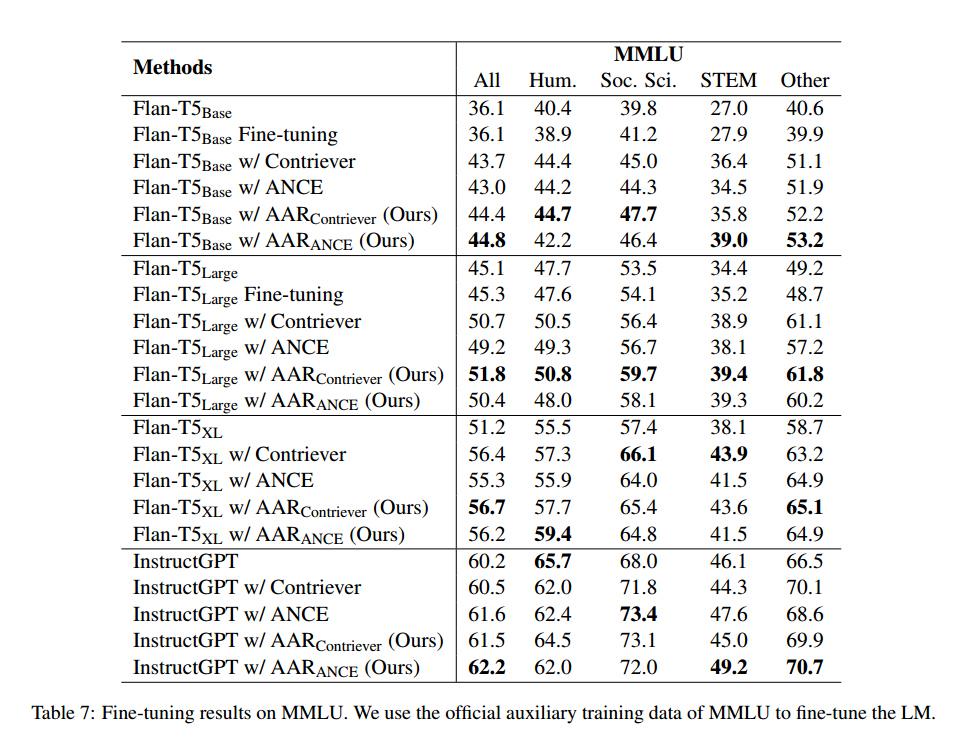
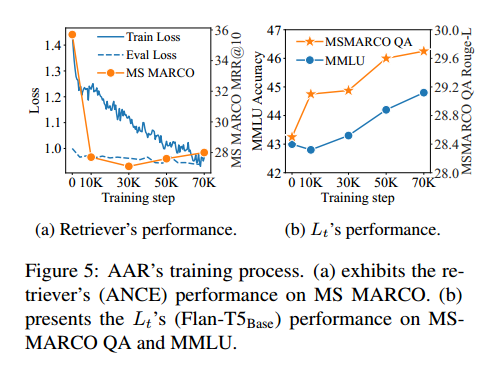


图5a和图5b分别绘制了检索器和LM在增强适应训练中的表现。在训练开始时，检索器在MS MARCO上的MRR@10显著下降，表明人类偏好的文档和LM偏好的文档之间存在很大的分布差距。随着检索器训练量和开发损失的不断下降，检索增强LM在MSMARCO QA上的表现逐渐提高，最终在MMLU上表现更好。这一结果表明，不同任务上的LMs可能具有共同的偏好，使得AAR可以很好地从单一源任务推广到异构目标任务。

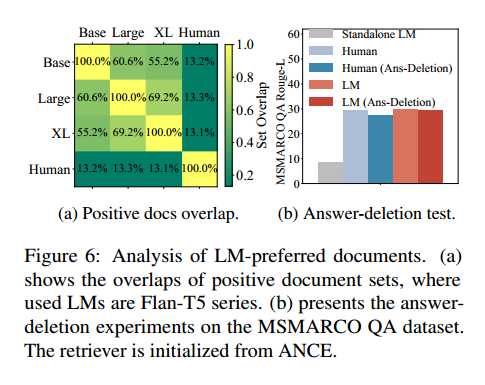


## LM偏好文档分析

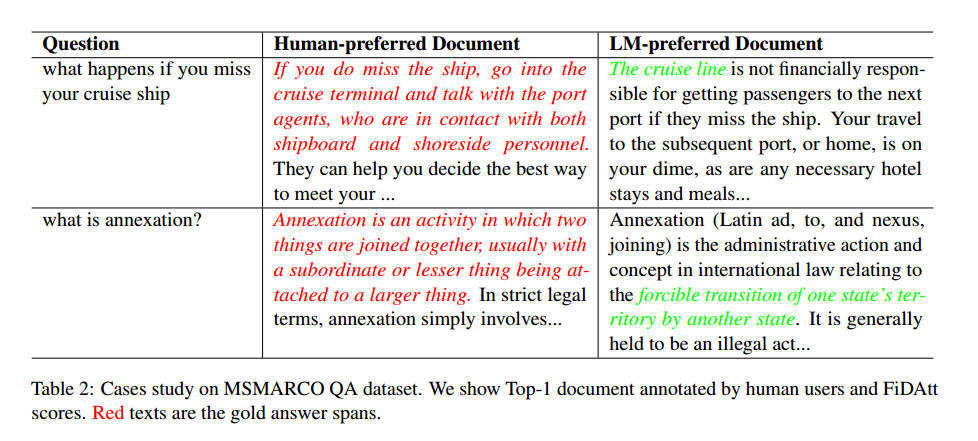
通过比较搜索用户和LMs之间的偏好文档，作者强调了使现有检索器适应LMs的必要性。一般来说，作者发现LM偏好文档可以从其他角度帮助LM，而不是搜索用户喜欢的完整信息。

定义两个正文档集和之间的重叠率为:

如图6a所示，人类用户注释的正面文档()集与LMs标注的正面文档集()重叠率很低，表明它们在选择有价值的文档方面有明显的倾向。相反，不同LMs之间的重叠度相对较高(超过55%)。这一证据为AAR的泛化能力提供了强有力的理论依据，因为不同大小的LMs倾向于注释相似的正面文档。此外，大小更接近的LMs通常具有更高的重叠。这意味着AAR对容量接近源LM的LM有更好的泛化能力。这些发现进一步验证了图4b中所示的结果。



为了深入分析人类偏好文档和机器偏好文档的不同之处，作者在表2中展示了从MSMARCO QA中抽取的两个代表性案例。人类偏好的文档总是可以在文本的开头提供黄金答案，而LMs偏好的文档可能不包含确切的答案。



然而LM偏好文档可以：

(1)提供一个新的视角来回答给定的问题，例如，“如果你错过了你的游轮，邮轮公司的责任”;

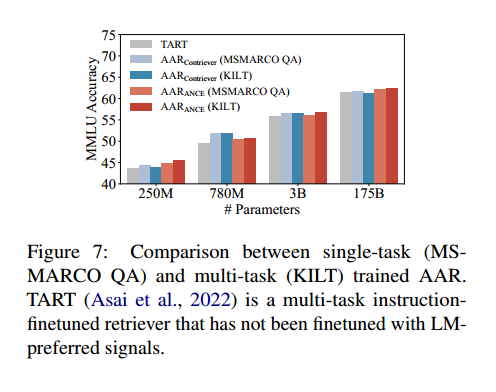
(2)给出一个具体的解释，而不是一个抽象的定义，例如，“一个国家的领土被另一个国家强行转移”。这些特征不同于搜索用户想要完整的信息，可以进一步帮助LMs进行基于知识的推理。

作者通过答案删除测试(即，从检索的文档中删除确切的答案范围)进一步检查LMs偏好文档的独特特征。如图6b所示，通过人类偏好文档训练的检索器(即human preferred检索器)或LM 偏好文档训练的检索器(即LM preferred检索器)都可以帮助LM回答给定的问题。然而，在答案删除后，LM preferred 检索器相较于human preferred 检索器，其性能下降更为显著。尽管精确匹配的答案较少(LM偏好文档为0.6%，而人类偏好文档为13.0%)，但是LM偏好文档从不同的角度提供有用的信息。因此，使用LM偏好文档调整检索器反过来可以使检索增强LM表现得更好。

## ARR的多任务训练

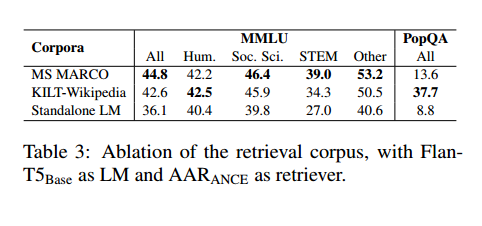
作者选择KILT作为作者的多任务数据源，它由5个类别(事实检查，实体链接，槽填充，开放域QA和对话)组成。每个类别取一个代表性的子任务，形成多个源任务的混合。

从图7可以看出，使用多任务KILT训练的ANCE可以持续优于单任务MSMARCO QA的ANCE，证明了多任务增强适应训练带来的更好的泛化能力。对于不同的任务，LMs可能会在偏好文档上略有不同，而AAR可以在多任务训练的帮助下更顺利地切换到目标任务。Contriever不会从多任务训练中获得很大的好处。作者推测这是因为Contriever已经接受了多种数据增强格式的预训练，因此比ANCE更好地泛化到新的数据分布。与AAR相比，多任务指令微调检索器TART的整体表现更差，这突出了在多任务训练中拥有LM偏好文档的好处。关于源任务选择的更详细的分析见附录B。



## 检索语料库的效果

表3表明，无论检索语料库是什么，AAR都比独立的LM获得一致的、实质性的性能提升。



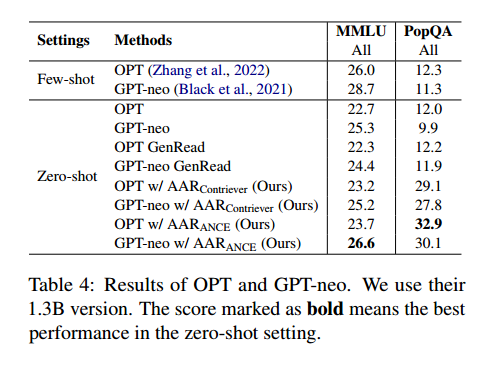
在MMLU上，使用MS MARCO作为检索语料库比KILT-Wikipedia更能提高LM。作者假设检索器已经接受了MS MARCO语料库的训练，因此具有更好的检索性能。

在PopQA上，如果使用MS MARCO作为检索语料库而不是KILT-Wikipedia，则模型性能将大幅下降。主要原因是PopQA数据集是从维基数据中抽取的，并且是为长尾问题设计的。部分长尾知识只能在KILT-Wikipedia中找到，而MS MARCO缺乏应该用于答案预测的不可或缺的证据。例如，给定“谁是梅丽莎·本恩的母亲?”， MS MARCO中没有包含答案“Caroline Benn”的文件。在这种情况下，为了利用AAR的能力，有必要将检索语料库与数据源对齐。

## ARR应用场景

为了检验AAR是否适用于缺乏zero-shot泛化能力的不可见LMs，作者还发布了OPT和GPTneo的结果。由于缺乏多任务指令调优，这些模型可能具有较差的zero-shot性能。

从表4中，可以发现AAR在MMLU上略微提高了两个LMs，而在PopQA上获得了显著的收益。作者推测，对于像PopQA这样的知识探测任务，LMs可以更容易地从检索增强中获益，在这种任务中，答案范围可以直接从检索到的文档中获得。MMLU要求LM不仅要理解检索到的证据片段，还要对它们进行基于知识的推理。OPT和GPT-neo在zero-shot下可能不具备这种能力。



总之，尽管AAR非常适合多任务指令微调的LMs，如Flan-T5系列和InstructGPT，但对于有时zero-shot性能很差的LMs，特别是在基于知识的推理方面，它可能不会带来显著的收益。作者认为多任务指令微调模型将成为未来工作的基础，因为它具有出色的zero-shot泛化能力，确保了AAR的广泛应用场景。

# 讨论

**LM偏好文档。**从LMs获取离散反馈信号是具有挑战性的，因为它需要卓越的标记能力，这不是LMs的设计目的。受ADist和Atlas的启发，作者利用FiDAtt分数来选择LM偏好的文档进行增强适应训练。然而，FiDAtt分数可能不能忠实地反映每个文档的实际贡献，因为LM可能更偏好关注可读而不是信息丰富的文档。此外，LM偏好文档的质量在很大程度上取决于检索增强LM的初始性能。

相似的工作通过计算检索器似然和LM似然之间的KL散度来训练检索器。然而，他们需要更大的源LM Curie (6.7B)来提供准确的LM似然信号。

在未来，强化学习可以作为训练检索器的替代方法，因为它通过直接利用LM的信号而不依赖于设计的规则来优化检索器。

**通用检索插件。**Chatgpt-retrievalplugin最近作为一个通用的检索插件在NLP社区中引起了关注。它从用户的数据源中检索最相关的文档，并定制ChatGPT的响应以满足他们的特定需求。作者相信AAR等技术基于检索到的信息增强黑箱ChatGPT将生成更合理响应的能力，从而促进以人为本的LM设计的发展。

# 总结和未来工作

本文介绍了通用检索插件，该插件利用通用检索器来增强可能事先未知或无法联合微调的目标LMs。作者建议的检索器AAR可以直接支持黑盒LMs，而不需要对LMs进行任何微调。这是通过以一个小的源LM提供的偏好文档和ground truth构建AAR的训练数据来完成的。

基于MMLU和PopQA的实证结果表明，AAR辅助的LMs在zero-shot场景下的性能大大优于独立的LMs，并且AAR可以很好地推广到不同大小和结构的LMs。分析结果表明：（1）LM偏好和人类偏好的文档具有互补性;（2）来自不同LMs的LM偏好文档明显重叠，具有相似大小的LMs倾向于产生更接近的文档集。

作者将更详细地解释不同的LMs如何与增强文档交互，并为将来的工作更合理地选择LM偏好的文档。作者希望他们的工作能够揭示出一种通用的方法，将大型LMs视为黑盒子，并使检索器适应以增强它们。

# 限制

由于计算资源的限制，作者没有对参数数为11B的Flan-T和参数数大于1.3 B的OPT进行评估。

由于OPT和GPT-neo在zero-shot设置中表现不佳，并且对于decoder-only模型来说，在分离每个输入文档的注意力分数是繁琐的，因此作者选择不使用它们作为源LMs。然而，作者证明了将encoder-decoder模型Flan-T作为他们的源LM对于decoder-only模型也具有鲁棒性。作者将探索新的方法，根据decoder-only模型的固有信号来注释LM偏好文档。

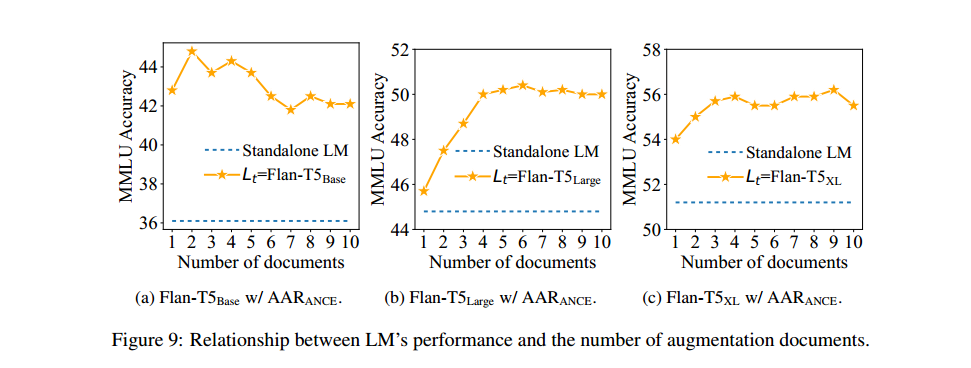
# 一些附录部分

## A.2增强文档的数量

面对不同目标任务的不同大小的LMs，可能需要无限数量的增强文档才能达到最佳性能。对于MMLU，作者分析了增强文档的数量如何影响lm的性能。

如图9所示，作者发现容量较大的LMs通常从更多的增强文档中获益更多。一种可能的解释是，较大的LMs更有能力集成来自多个文档的信息，并基于它们执行复杂的推理。

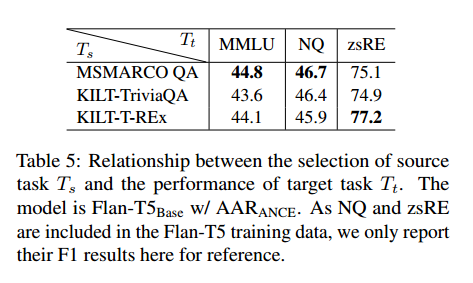
对于PopQA，使用3个增强文档可以在所有lm中实现最佳性能。



## 源任务的选择

作者提供源任务的详细选择，使用各种源任务和目标任务进行分析。MSMARCO QA、KILT-TriviaQA和NQ属于开放领域 QA，而KILT-T-REx和zsRE属于Slot Filling。MMLU属于多任务语言理解，在任务目标方面更接近开放领域QA。

如表5所示，将源任务的类别与目标任务对齐时，w/ AAR的LM通常可以获得最佳结果。作者猜想这是因为LM可能对来自同一数据集类别的任务共享类似的偏好文档，从而使AAR更容易泛化。此外，以MSMARCO QA作为源任务在MMLU上的性能最好。这验证了在作者的主要实验设置中将设置为MSMARCO QA的合理性。



# 补充说明

## 检索增强生成-RAG（Retrieval Augmented Generation）

参考文章：[一文搞懂大模型RAG应用（附实践案例） - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/668082024)

大致流程：

数据准备阶段：数据提取——>文本分割——>向量化（embedding）——>数据入库

应用阶段：用户提问——>数据检索（召回）——>注入Prompt——>LLM生成答案

## 检索器（Retrievers）

参考文章：[【LangChain】检索器(Retrievers)\_langchain 中retrievers是什么-CSDN博客](https://blog.csdn.net/u013066244/article/details/131843772)

检索器是一个接口，它根据非结构化查询返回文档。它比向量存储更通用。检索器不需要能够存储文档，只需返回（或检索）它。向量存储可以用作检索器的主干网络，但也有其他类型的检索器。

## 主干网络（backbone）

大多时候指的是提取特征的网络，其作用就是提取图片中的信息，共后面的网络使用。

## log-liklihood

参考文章：[对数似然函数值/最大近然估计/log likelihood\_对数似然估计-CSDN博客](https://blog.csdn.net/wydbyxr/article/details/83212703)

 最大似然估计法的思想很简单：在已经得到试验结果的情况下，作者应该寻找使这个结果出现的可能性最大的那个X作为真X的估计。求X的极大似然估计就归结为求L(X)的最大值点，而由于对数函数是单调增函数，所以对L(X)取log。

对log（L(X)）关于X求导数，并命其等于零，得到的方程组称为似然方程组。解方程组log（L(X)），又能验证它是一个极大值点，则它必是L(X)的最大值点，即为所求的最大似然估计。

## 知识蒸馏

参考文章：[一分钟带你认识深度学习中的知识蒸馏 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/258390817)

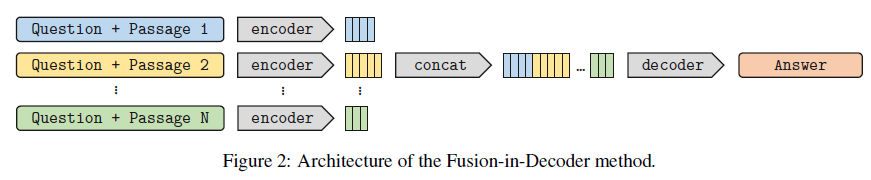
## Multi-hop

参考文章：[浅谈多跳阅读理解 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/133483274)

## Zero-shot

参考文章：[零样本学习(Zero-Shot Learning)简介与分类 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/436720853)

## Fusion-in-decoder

参考文章：[Fusion-in-Decoder (FiD) 简读 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/564043932)

## FiD attention score

参考文章：[ICLR-2021|Distilling Knowledge from Reader to Retriever for Question Answering - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/576171199)

部分原文：检索器标签的具体计算方法为：给定问题*q*和对应的支持文档集*，*取**decoder**中**每一层**的**每一个注意力头**的**第零个token**的**query向量Q**，与**encoder拼接序列**上的**某一个检索段落**对应的**每一个token**的**key向量K**，计**算Q与K**之间的点积的softmax值的平均作为检索段落  的注意力得分***。***

## 近似最近邻搜索算法(approximate nearest neighbor search)

参考文章：[近似最近邻搜索算法 ANNOY（APPROXIMATE NEAREST NEIGHBORS OH YEAH） - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/454511736)

## Crosss-attention

参考文章：[【科研】浅学Cross-attention？\_cross attention-CSDN博客](https://blog.csdn.net/MengYa_Dream/article/details/126688503)

[Self -Attention、Multi-Head Attention、Cross-Attention\_cross attention-CSDN博客](https://blog.csdn.net/philosophyatmath/article/details/128013258)

## KL-divergence

参考文章：[机器学习\_KL散度详解（全网最详细）\_kl散度计算公式-CSDN博客](https://blog.csdn.net/Rocky6688/article/details/103470437)

## 最邻近负对比估计(ANCE)

参考文章：[Approximate Nearest Neighbor Negative Contrastive Learning for Dense Text Retrieval-CSDN博客](https://blog.csdn.net/doyouseeman/article/details/118918909)

## Hard Negative Samples

参考文章：[难负例 多模态：对比学习Hard Negative Samples论文小结 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/679594639)

## FLOPs

参考文章：[CNN 模型所需的计算力flops是什么？怎么计算？ - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/137719986)

## MRR

参考文章：[推荐系统常用的评价指标：HR、NDCG、MRR\_hr@10 和 ndcg@10是什么-CSDN博客](https://blog.csdn.net/shiaiao/article/details/109004341)

[评价指标reacll@10,mrr@10,ndcg@10,hit@10的含义-CSDN博客](https://blog.csdn.net/wcc8848/article/details/123117982)

## 槽填充(Slot -Filling)

参考文章：[槽填充（Slot Filling）的定义、用途、意义及其他 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/93853430)