Adapting Language Models for Zero-shot Learning by Meta-tuning on Dataset and Prompt Collections

通过对数据集和提示集合进行元调整来适应零样本学习的语言模型

# abstract摘要

大型预训练语言模型 (LM)，例如 GPT-3，已经获得了执行零样本学习的惊人能力。例如，为了在没有任何训练示例的情况下对情感进行分类，我们可以用评论和标签描述“用户喜欢这部电影吗？”来“提示”LM，并询问下一个词是“是”还是“否”。然而，下一个词预测训练目标仍然与目标零样本学习目标不一致。为了解决这个弱点，我们提出了元调整，它通过在一组数据集上微调预先训练的语言模型来直接优化零样本学习目标。我们专注于分类任务，并通过聚合 43 个现有数据集并以问答 (QA) 格式注释 441 个标签描述来构建元数据集。在对未知任务进行评估时，元调整模型的性能优于相同大小的 QA 模型和之前基于自然语言推理的 SOTA zeroshot 学习系统。此外，将参数数量从 220M 增加到 770M 将 AUC-ROC 分数提高了 6.3%，我们预测更大的模型会表现得更好。因此，在开箱即用的语言模型上衡量零样本学习性能可能会低估它们的真正潜力，而社区范围内在聚合数据集和统一格式方面的努力可以帮助构建更好地回答提示的模型。

# 1 Introduction

零样本分类 (ZSC) 的目标是在没有任何示例的情况下使用标签描述对文本输入进行分类（Yin 等，2019）。大型语言模型——其唯一的训练目标是在给定上下文的情况下预测下一个单词——已经获得了执行 ZSC 的惊人能力（Radford 等人，2019 年；Brown 等人，2020 年；Le Scao 和 Rush，2021 年）。例如，要对句子“This movie is Amazing！”是否为正词进行分类，我们可以通过上下文“Review: This movie is Amazing! Positive Review? \_\_\_”来提示语言模型，并检查下一个单词是否更有可能是“是”或“否”（Zhao et al., 2021）。为了将 ZSC 转换为 LM 模型可能表现良好的语言建模 (LM) 任务，最近的许多工作都集中在寻找更好的提示上（Shin 等人，2020；Schick 和 Schütze，2020a,b；Gao 等人， 2021）。

然而，LM 训练目标是相关的，但仍然与回答提示的目标目标不一致。我们的工作通过微调直接优化零样本分类目标来解决这个弱点（第 4 节）。这要求我们 1) 将不同的分类任务统一为相同的格式，以及 2) 收集分类数据集和标签描述（提示）的集合以进行训练（第 2 节）。由于我们在元数据集上微调我们的模型，因此我们将我们的方法命名为元调整。

我们专注于二元分类任务并将它们统一为“是”/“否”QA 格式（Clark 等人，2019 年；McCann 等人，2018 年），其中输入作为上下文提供，并提供标签信息在问题中（图 1（a））。使用这种格式，我们从 Kaggle、SemEval、HuggingFace 和其他论文中列出的 43 个不同来源收集了一组多样化的分类数据集。这些任务的范围从仇恨言论检测、问题分类、情感分类到立场分类等，类型从教科书、社交媒体到学术论文等。这些数据集总共包含 204 个唯一标签，我们手动标注了 441 个标签说明（图 2）。

为了评估 ZSC，我们需要定义模型在训练期间未见过的任务。虽然先前的工作通过禁止在训练期间出现相同的标签或相同的数据集来考虑“看不见”的不同概念，但我们的工作通过禁止类似的数据集更严格地定义“看不见”。例如，我们考虑 AG News 主题分类数据集（Zhang et al., 2015）和 Yin 等人的主题分类数据集。 （2019）相似，即使它们的来源和标签空间不同。

对于大多数标签，元调整比 UnifiedQA 改进了 ZSC（图 1（c））。此外，更大的模型更好，因此我们预测元调整将适用于更大的模型。我们还发现，通过在与测试数据集相似的数据集上进行训练、集成不同的标签描述或使用 QA 模型（第 5.1 节）进行初始化，可以稍微提高性能。我们所有的发现都在不同的稳健性检查下可靠地成立（第 5.2 节），并且我们的方法优于之前的 SOTA Yin 等人。 (2019) 使用相同的预训练方法（第 5.3 节）。

我们的结果表明了两个有希望的未来方向（第 6 节）。首先，大语言模型（例如 GPT-3）的零样本学习潜力，目前通过上下文提示来衡量，可能被广泛低估了；元调整可能会显着提高它们的性能。其次，社区范围内聚合和统一数据集的努力可以扩大零样本学习模型的培训和评估。然而，另一方面，元调整方法可能会激励 LM 推理 API 的提供者收集用户的提示，从而可能导致更大规模的安全、隐私和公平问题（A 部分）。

贡献 总而言之，我们 1) 使用专家标记的标签描述来管理分类数据集的数据集。 2) 展示训练模型以执行零样本学习的简单方法，以及 3) 确定提高性能的几个因素；特别是，较大的预训练模型更好。

# 2 Data

我们收集了广泛的分类数据集，并将它们统一为“是”/“否”问答格式以进行二元分类。然后我们将相似的数据集组合在一起，以确定在评估期间哪些是未见过的任务。

收集分类数据集我们从 Kaggle2、Huggingface（Wolf 等人，2020）、SemEval3 和其他论文中收集分类数据集。我们查看了这些来源，只考虑了英语分类数据集。我们还跳过了我们认为已经由我们集合中的其他数据集更好地代表的任务。然后我们手动检查了每个剩余数据集中的一些示例，以确保它看起来似乎很干净。

这些分类数据集的目标包括但不限于情感分类（IMDB Reviews, Maas et al. (2011a)）、主题分类（AG News, Zhang et al. (2015)）、语法判断（CoLA, Warstadt et al. (2015)） al. (2018))、释义检测 (QQP4)、定义检测 (SemEval 2020 Task 6, Spala et al. (2019))、姿势分类 (SemEval 2016 Task 6, Mohammad et al. (2016)) 等。体裁包括学术论文、评论、推文、帖子、消息、文章和教科书。数据集的完整列表在附录 B 中。总的来说，我们的目标是通过建立更广泛的研究社区的研究来实现高度多样化的任务和类型。我们的方法是对 Weller 等人的方法的补充。 (2020)，它要求 turkers 生成任务，以及 Mishra 等人的。 （2021），它通过分解用于构建阅读理解数据集的现有模板来生成任务。布拉格等人的并发工作。 (2021) 统一了小样本学习的评估；他们的零样本评估设置与我们的最接近，他们使用模板和语言表达器（Schick 和 Schütze，2020a）来指定任务的语义。

我们的一些数据集有噪声且未经同行评审，或包含对 ZSC 而言过于复杂的任务（例如 Multi-NLI，Williams et al.(2018)）。不测试。我们在第 5 节中运行我们的实验之前做出这些决定，以防止选择偏差。

统一数据集格式我们将每个分类数据集转换为“是”/“否”问答格式，并在问题中提供标签信息。对于每个标签，我们注释1-3个问题。如果标签为空（例如，不在情感分类数据集中表达特定情感），我们跳过这个标签。三位作者3手动注释了 204 个独特标签的 441 个问题，每个问题

至少由另一位作者校对。具体示例见图 2，一些代表性标签说明见图 3。

此外，一些数据集包含数千个标签（Chalkidis 等人，2019 年；Allaway 和 McKeown，2020 年）。在这种情况下，我们使用模板自动合成标签描述并将它们从评估中排除。

对相似数据集进行分组我们的目标是测试模型泛化到与训练任务足够不同的任务的能力。因此，在测试时，我们不仅需要排除元调整阶段出现的相同数据集，还需要排除相似的数据集。

这就提出了一个挑战：两个数据集是否执行相同的任务涉及主观意见，并没有普遍认同的定义。在一种极端情况下，大多数数据集可以算作不同的任务，因为它们具有不同的标签空间和输入分布。在另一个极端，所有数据集都可以被认为是同一个任务，因为它们都可以统一为问答格式。

为了应对这一挑战，我们创建了一组标签，每个标签都描述了一个数据集属性。标签集包括领域分类、文章、情感、社交媒体等，它们的完整集可以在附录 C 中看到。 然后我们定义

如果两个数据集与同一组标签相关联，则它们是相似的，并禁止模型向其中一个学习并在另一个上进行测试。例如，我们的工作考虑了 Zhang 等人的主题分类数据集。 (2015) (AG News) 和 Yin 等人。 (2019) 相似，因为它们都对文章的主题进行分类，即使它们的来源和标签空间不同。一些示例数据集组可以在图 4 中看到。

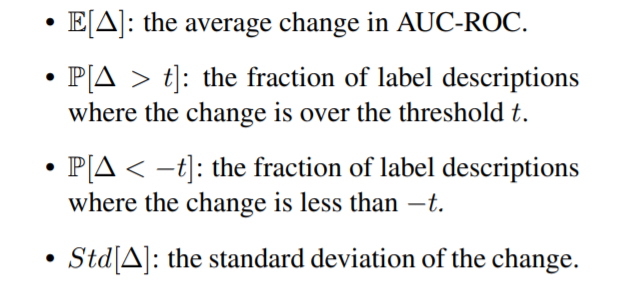
尽管如此，我们的程序并不是万无一失的，人们可以争辩说，我们对看不见的任务的概念虽然比以前的工作更苛刻（Yin 等人，2019 年；Pushp 和 Srivastava，2017 年），但仍然是宽容的。因此，作为额外的稳健性检查，对于我们评估的每个数据集，我们手动识别并列出附录 F 中训练期间允许的最相关数据集。例如，与 IMDB 评论情感分类数据集最相关的数据集是 Yin 等人的情感分类数据集。 (2019)，将输入文本分为 9 种情绪，如“喜悦”、“惊讶”、“内疚”等。我们认为情绪分类化数据集是相关的，因为情感分类通常涉及识别情感。 然而，人们也可以说它们是不同的任务：它们的输入和标签空间不同，悲伤可能是由一场巨大的悲剧引起的，或者是一部浪费用户时间的烂电影。 按数据集相似性分组的标签描述的完整列表在附录 D 中。

我们总共花费了大约 200 个小时来收集这个数据集。 这个时间估计包括浏览数据集 repos 和最近的 NLP 论文，编写程序来下载数据集并统一它们的格式，注释标签描述，执行质量控制，并记录收集过程。

# 3 Metrics

为了可靠地汇总不同数据集的性能并提供尽可能多的信息，我们报告了一组描述性统计数据，并在比较两个模型时提供可视化效果。 我们通常不会将模型在不同数据集上的表现减少为一个标量，而仅比较这个数字。

描述性统计 对于每个标签描述（问题），我们通过将“是”答案视为正类来计算 AUC-ROC 分数 6。在计算每个标签的 AUC-ROC 分数后，我们计算以下一组描述性统计数据 比较两个模型。假设模型 Y 假设比 X 更好。将 ∆ 表示为标签描述的 AUC-ROC 从 X 到 Y 的变化，我们可以总结 ∆ 在标签描述集合中的分布情况，统计如下：



在主要论文中，我们在此分布中对每个标签描述进行平均加权以计算上述统计量。 我们也可以对每个标签或数据集进行平均加权，相应的结果在附录 E 中。为了确保我们的结论是稳健的，我们认为只有当 E[Δ] > 0 和 P[Δ > t] > P[∆ < -t] 对于所有 t ∈ {1%, 5%, 10%}，在所有三种加权类型下。 换句话说，我们声称只有当 12 个条件同时成立时，一个模型才比另一个更好。

可视化我们使用散点图来可视化和比较两个模型的性能，其中每个点代表一个标签描述，它的 x 值代表模型 X 的 AUC-ROC 分数，它的 y 值代表 Y 的值。 如果大多数点都在身份线 y = x 之上，则模型 Y 优于 X。

描述性统计和可视化在图 5 中进行了解释.

# 4 Model

架构 我们以与 UnifiedQA (Khashabi et al., 2020) 相同的方式格式化模型的输入，它将上下文连接到问题并在两者之间添加一个“[SEP]”标记。然后我们将连接的输入输入到T5 编码器并通过对第一个解码标记的“是”/“否”概率进行归一化来产生答案分数。除非另有说明，否则我们使用 T5-Large（7.7 亿个参数）初始化我们的模型。我们有时会与UnifiedQA 模型 (Khashabi et al., 2020)，在广泛的问答数据集上训练。为了公平比较，我们也使用了用 T5-Large 初始化的 UnifiedQA 模型。对非 Seq2Seq 预训练模型进行元调整，例如 BERT (Devlin et al., 2019) 或 RoBERTa (Liu et al., 2019)，我们在池化输出/“[CLS]”标记之上添加了一个 MLP 层，以在“是”/“否”之间进行分类”。我们将模型架构（Ye 和 Ren，2021；Li 和 Liang，2021；Lester 等，2021）和训练目标（Murty 等，2021；Yin 等，2020）的改进留给未来的工作。

元调整我们创建了一个训练分布，在数据集、标签描述和“是”/“否”答案之间取得平衡。为了创建用于元调整的下一个训练数据点，我们从训练分割中随机均匀地选择一个数据集（u.a.r.）；然后我们选择一个标签描述（问题）u.a.r.并以 50% 的概率选择答案为“是”/“否”的文本输入。为了防止过度拟合，我们不会对标签描述和文本输入的任何组合进行两次训练。除非另有说明，我们对模型进行了 5000 步元调整并使用批量大小为 32。我们没有调整任何超参数或训练配置，因为它们在我们的第一次尝试中运行良好。为了评估每个数据集上的 ZSC 性能，我们将一组类似的数据集作为评估集省略，并在其余数据集上进行训练。在 Quadro 8000 上，实验总共需要大约 250 个 GPU 小时.

# 5 Results

## 5.1 Hypotheses and Conclusions

我们调查并验证了以下假设，按重要性降序排列。

• 元调整模型在零样本分类中优于一般问答模型。

• 较大的预训练模型更好。

• 预训练完成繁重的工作。

• 性能可以通过对类似数据集进行训练、使用 QA 模型进行初始化或集成标签描述来提高。

• 提前停止对性能至关重要。

元调整模型更好。我们将元调整的 T5-Large 模型（770 M 参数）7 与开箱即用的相同大小的 UnifiedQA 模型（Khashabi 等人，2020）进行了比较。相关描述性统计见表1第一行和图6(a)。为 ZSC 调整模型将平均 AUC-ROC 提高 3.3%。

较大的预训练模型更好。我们将 T5-Base（2.2 亿个参数）与 T5-Large（7.7 亿个）进行了比较。统计数据可以在表 1 的第二行和图 6 (b) 中看到。将模型大小从 220 M 增加到 770 M，平均 AUC-ROC 提高了 6.3%。

预训练完成了繁重的工作。在图 (c) 和表 1 的第三行中，我们比较了预训练和随机初始化，其中后者无法击败随机基线（平均 AUCROC 0.503）。因此，仅靠元调整远不能使模型执行 ZSC。一种直观的解释是，模型在 LM 目标下进行预训练后已经“知道”如何执行 ZSC，并在元调整期间学习如何使用这些知识。

对类似数据集进行训练可提高性能。与以前不同，我们不再避免对来自同一组的类似数据集进行训练。相反，我们执行直接的留一法交叉验证。统计见表1第四行和图6(d)，平均AUC-ROC提高了0.7%。性能提升并不像增加模型大小或适应 ZSC 那样显着。我们推测是因为我们没有收集足够的数据集；否则，可能会有更多相似的数据集，从而提高 ZSC 性能。

集成标签描述可提高性能。与其针对每个标签向模型询问单个问题并获得答案为“是”的概率，我们可以通过询问具有相同含义的多个问题获得的概率进行平均。这种方法不同于传统的集成，后者通常需要存储/训练多个模型以求平均。表 1 的第五行和图 6 (e) 验证了集成描述略微提高了性能（0.7% AUC-ROC 分数）。

使用 UnifiedQA 进行初始化可提高性能。图 6 (f) 和表 1 的第六行将 UnifiedQA 与 T5 初始化进行了比较。使用 UnifiedQA 初始化将平均 AUC-ROC 提高了 1.1%。

提前停止对性能至关重要。如果我们训练模型的时间过长，模型可能会简单地“记住”某些标签描述对应于某些训练任务，而在看不见的任务上的性能可能会下降。为了探索这种可能性，我们对模型进行了 10 万步元调整，这是我们默认设置的 20 倍，并鼓励模型记住训练任务。然后我们在 Yin et al. (2019) 的三个基准零样本分类数据集上评估它们（我们在下一节）我们计算了所有 3 个数据集的所有标签描述的平均 AUCROC，并将它们绘制在图 7 中。

随着训练的继续，性能下降 8。然而，另一方面，AUC-ROC 3% 的性能下降并不是致命的，模型的性能仍然比随机猜测好得多。

## 5.2 Robustness Checks

我们检查了一系列额外的结果，以确保我们的结论是可靠的。表 1 和图 6 中观察到的改进可能是由于少量标注了更多描述的标签的改进，或者是具有更多不同标签的数据集的改进。附录 E.1 通过为每个标签/数据集分配相等的权重来比较性能。

为了为我们的预测提供更多支持证据，即更大的模型更好，附录 E.2 将 60M 参数模型与 220M 参数模型进行了比较，发现后者要好得多。然而，一个问题是我们的模型是用 T5 初始化的（Raffel 等人，2019 年），它在开放网络上训练并且可能已经看到了我们收集的数据集。因此，更大的模型可能更好，因为它们更擅长记忆（Sagawa 等，2020）。附录 E.3 解决了这个问题，它表明更大的模型也能更好地使用 BERT 初始化（Devlin 等人，2019 年），这是在维基百科和图书语料库（Zhu 等人，2015 年）上训练的。我们还在附录 G 中报告了模型在每个数据集上的性能以供读者参考。

## 5.3 Comparison with Yin et al. (2019)

本节表明我们的方法比 Yin 等人构建的零样本分类系统具有更高的性能。 (2019)。他们的系统集成了几个基于 RoBERTA-Large（3.55 亿个参数，Liu 等人（2020））的自然语言推理模型，以及另一个训练用于对维基百科文章进行分类的模型。它在三个分类数据集上进行了评估：

• 主题（10 向）：对文章领域进行分类，例如家庭与关系、教育、体育等。度量标准是准确性。

• 情绪（10 向）：对情绪类型进行分类，例如喜悦、愤怒、内疚、羞耻等。度量是标签加权的 F1。

• 情况（12 向）：对灾害情况进行分类，例如政权更迭、犯罪和暴力，以及他们需要的资源，例如搜救。度量是标签加权的 F1。

我们使用与 Yin 等人完全相同的评估指标。 (2019)，当模型对多标签分类回答“是”9时，采用相同的标签解析策略。具体来说，当模型在多个标签上预测“是”时，选择概率最高的一个。为了公平比较，我们对相同大小的 RoBERTa 进行元调整，并将其与 Yin 等人中性能最高的模型进行比较。 (2019) 对于三个数据集中的每一个。

结果如表 2 所示，我们的模型在使用相同的预训练方法的所有 3 个数据集上都具有更高的性能。

# 6 Discussion and Future Directions

主要内容我们构建了一个分类数据集，以通过元调整使语言模型适应零样本分类。适应模型优于通用问答模型和基于自然语言推理的现有技术。我们预测元调整在更大的模型上会更有效，而当前零样本学习的工程上限可能被广泛低估。

聚合和统一数据集我们研究的主要瓶颈是手动收集范围广泛的数据集并统一它们的格式。困难在于：1）我们需要集思广益并广泛审查 NLP 文献，以决定要寻找哪些新任务； 2）不同的数据集以不同的格式对它们的数据进行编码，我们需要为每个数据集手动编写程序以转换为所需的格式； 3) 很难仅仅通过数据集的来源来判断数据集的质量，有时我们需要手动检查数据集。如果我们作为一个社区可以更好地聚合和统一数据集，我们就有可能在更大范围内训练和评估零样本学习模型。

元调整作为一种探索 测量 GPT-3 等大型语言模型的智力（Hendrycks 等人，2021a，b）或少样本学习能力（Brown 等人，2020 年）的兴趣越来越大。然而，由于这些模型不适用于回答这些提示（Holtzman 等人，2021 年），我们怀疑其执行小样本学习的知识和真正潜力比报告的要高得多。由于预训练完成了繁重的工作，元调整不太可能为模型提供额外的 ZSC 能力，因此我们可能首先使用元调整作为探针，使它们在测量其性能之前适应回答提示。

尽管如此，为了使这种方法变得严谨，解释和控制探针的强度将是一个重要的未来方向（Hewitt 和 Liang，2019）。例如，如果训练集包含与待测试的提示太相似的提示，则探测将毫无意义。

超越浅层相关性 一种可能性是该模型仅从元调整中学习浅层统计相关性，而不是“更复杂的推理技巧”。例如，“令人兴奋”一词可能更多地出现在正面评论中。这不太可能，因为较大的模型始终比较小的或随机初始化的模型更好。为了解释这种性能差距，较大的模型必须学会在元调整期间使用更复杂的特征。

与元/多任务学习的关系我们的方法与元学习 (Yin, 2020; Murty et al., 2021) 和多任务学习 (Ye et al., 2021; Aghajanyan et al., 2021）。元学习和多任务学习通常至少涉及来自目标任务的几个例子；然而，在我们的设置中，模型不会从任何目标任务示例中学习。我们名字中的“元”并不意味着“元学习”，而是反映了我们的模型从任务的元数据集中学习的事实。

尽管如此，我们的框架可以很容易地适应少量学习设置，这使语言模型能够学习从上下文示例中学习（见下文）。由于这种方法将学习过程建模为序列分类问题，因此可以将其视为类似于 (Ravi and Larochelle, 2016) 的一种元学习形式。

注释提示 我们的三位作者对标签描述进行了注释。由于他们都是了解机器学习和自然语言处理的计算机科学专业学生，因此他们可能无法代表此 ZSC 应用程序的最终用户群。标注匹配目标用户分布的提示将是一个重要的研究方向。

此外，更短、更自然的描述有时无法准确捕捉标签的语义。例如，在尹等人。 （2019），“医疗”标签的描述是“人们需要医疗援助”；或者，它可以更长但更准确：“人们需要支持医生和其他卫生专业人员工作的联合卫生专业人员”。如何在没有专家的努力下可扩展地生成更准确和详细的标签描述将是另一个未来方向。

优化提示我们的工作是对最近优化提示以实现更高准确性的工作的补充。即使我们的元调谐模型专门用于回答提示，它对不同提示的反应仍然可能非常不同。例如，在立场分类数据集 (Barbieri et al., 2020) 中，我们为同一个标签标注了两个标签描述（提示）：“这篇文章支持无神论吗？”和“这篇文章反对有宗教信仰吗？”。它们具有相似的含义，但前者的准确性比后者低得多。我们推测这是因为该模型无法为“无神论”等抽象概念提供依据。

其他扩展 我们推测元调整可以扩展到零样本二元分类之外的更多样化的任务。 为了扩展到多标签分类，当模型对多个标签预测为正时，我们需要开发一个程序来解析标签。 为了扩展到少样本学习，我们需要增加上下文长度以将几个训练示例适合输入，这需要更大的上下文窗口，因此需要更多的计算资源。 为了扩展到其他序列生成任务，我们需要收集各种不同的序列生成任务来对模型进行元调整，例如机器翻译、摘要、自由形式的问答、语法校正等。