arXiv:2602.01237v1 [cs.AI] 1 Feb 2026

大语言模型高效推理时推理的预测性调度

KatrinaBrown* **Aneesh Muppidi***哈佛学院 哈佛学院
katrinabrown@college.harvard.edu
aneeshmuppidi@college.harvard.edu**Rana Shahout**哈佛工程与应用科学学院rana@seas.harvard.edu

摘要

大语言模型通过生成多条思维链（CoT）轨迹，在复杂推理任务上取得最先进的准确率；然而，对每个查询使用固定的令牌预算，会导致简单输入计算过度、困难输入计算不足。我们提出预测性调度，这是一种即插即用框架，可在进行完整生成之前，预先运行轻量级预测器——基于中间 Transformer 隐藏状态的多层感知机（MLP），或针对原始问题文本进行 LoRA 微调的分类器——以估计每个查询的最优推理长度或难度。我们的贪心批次分配器会在各查询之间动态分配固定的总令牌预算，以最大化预期准确率。在 GSM8K 算术基准测试上，与统一预算分配相比，预测性调度在词元成本相同的情况下，准确率绝对提升最高达 7.9 个百分点，弥补了与拥有完美预见能力的预言机之间超过 50% 的差距。系统的逐层研究显示，Transformer 的中间层（第 12 至 17 层）包含最丰富的规模估算信号。这些结果表明，预先进行预算预测能够精细调控计算量–准确率权衡，为构建延迟敏感且成本高效的大语言模型部署提供了切实可行的路径。

项目网站与代码： https://aneeshers.github.io/predictive-scheduling/

1 引言

现代大语言模型（LLM）具备卓越的思维链（CoT）推理能力，为实时代码自动补全、交互式辅导和决策支持等应用提供了动力。然而，推理时成本仍是一个关键瓶颈：对每个查询采用固定的令牌预算，要么在简单输入上浪费令牌，要么导致困难输入的预算不足，从而造成不必要的延迟并推高云账单——这是生产环境大语言模型服务和端侧部署所关注的关键问题。

我们首先探讨每查询预算的异质性这一问题，具体而言：能否在生成开始前预测每条推理轨迹所需的令牌数量，或每个查询的难度，从而指导预算分配？以往方法要么依赖少样本启发式方法来猜测单个最优轨迹长度 [Han 等人， 2024], 在生成过程中进行单检查点提前终止检查[Li 等人，2024年，Fu 等人，2024年a], 或依据队列长度、历史运行时间等表层信号来安排批次 [Liu 等人，2023年，Damani 等人， 2024]。这些方法都没有利用 Transformer 隐藏状态丰富的内部特征，来学习运行前对所需轨迹长度和查询难度进行细粒度估计，也没有在固定总令牌预算下将这些估计纳入全局批次分配器。

具体而言，我们提出了五项贡献

*贡献相同。

1. 我们提出了 2 个轻量级预测器，将 Transformer 隐藏状态映射为推理轨迹达到正确答案所需的词元长度 估计值。2. 我们开发了一种轻量级难度分类器，结合少样本提示和 LoRA 微调，在生成前将查询分为简单、中等或困难。3. 我们进行了系统的逐层分析，发现中间 Transformer 层（12–17）对推理长度估计提供了最强的预测信号。4. 我们设计并实现了一种贪心批次分配算法，在固定总预算下动态为每条推理轨迹分配词元预算，以最大化预期准确率提升。5. 我们在 GSM8K 算术推理基准上证明，与非自适应调度相比，我们的组合式预测调度在词元成本相同的情况下，准确率最高可绝对提升 7.9%，并弥合了与具备完美长度和难度估计的预言机之间超过一半的差距。

我们的预测式调度框架为对延迟和成本敏感的大语言模型部署提供了一种实用的即插即用方法，表明在运行前预测预算和难度，无需修改底层语言模型，即可显著提升推理效率。

2 相关工作

2.1 思维链与元解码

大语言模型通过生成中间的“思考”步骤来提升多步推理能力。思维链提示将问题拆解为多个子步骤，以提高算术和逻辑任务的准确率 [Wei 等人， 2022], 而自一致性解码则汇总多条采样得到的思维链，以降低方差和错误 [Wang 等人， 2022]。思维树进一步探索部分解答构成的树状结构，以引导搜索 [Yao 等人， 2023]。这些方法效果显著，但假设链长度固定或每个查询的预算一致，因此会产生大量额外计算。

2.2 批次调度与词元重排序

处理难度各异的请求时，简单的先进先出或统一批处理可能会遭遇队头阻塞，并在简单查询上浪费令牌。vLLM 调度器会预测剩余令牌数，对查询重新排序并进行抢占，以提高吞吐量 [Liu 等人， 2023], 相关的预调度技术则可降低多租户服务中的延迟 [Zhang 等人， 2021]。Dynasor 会在推理过程中的固定检查点，动态为判定为“困难”的查询分配额外计算 [Fu 等人，2024年a]。这些外部调度器虽然能改善平均延迟，但依赖表层启发式方法或单一检查点的决策，而非通过学习逐查询估计所需的推理长度。

2.3 Transformer 中的自适应计算

另一项研究路线会根据输入难度调整模型的内部计算。深度自适应 Transformer 会改变每个输入所执行的层数 [Elbayad 等人； 2020],而混合专家架构则将令牌路由至部分专家子模块，以节省计算量 [Fedus 等人； 2021]。这些方法会“即时”调整逐输入计算量，但无法在运行前提供逐查询长度估计，也无法在固定令牌预算下进行批次级调度。

2.4 逐查询令牌预算预测与贪心分配

提前估计每个查询的最优推理长度，可以避免浪费令牌和推理不足。Wu 等人分析了准确率如何随着思维链长度增加而先上升、随后趋于平稳或下降，并证明每个问题都存在最优思维链长度 [Wu 等人， 2025]。TALE 使用少样本提示和小型后训练模型，预测每个查询的最优令牌预算[Han 等人， 2024]。Fu 等人通过学习排序训练长度排序预测器，在批处理前按请求的相对长度排序 [Fu 等人，2024年b]。Li 等人提出一种学习得到的早停标准，在生成过程中判断增加推理链是否可能提升正确性 [Li 等人，2024]。Damani 等人训练轻量级预测器（多层感知机或 LoRA），以估计增加更多推理轨迹所带来的边际奖励，并采用贪心算法，在

固定预算内 [Damani 等人， 2024]。尽管这些方法能够预测或调整令牌用量，但它们并未利用大语言模型隐藏状态中丰富的内部特征，也未系统地识别哪些 Transformer 层包含最强的预测信号。

2.5 课程学习与难度感知批处理

课程学习提出按由易到难的顺序组织训练数据，以加快收敛并提升泛化能力 [Bengio 等人， 2009]。自定进度学习拓展了这一思路：利用随训练进展而变化的隐式难度指标，同时学习课程安排与模型参数 [Kumar 等人， 2010]。理论和实证研究证实，设计良好的课程能够改变优化景观，从而加快收敛，同时不改变全局最优解 [Hacohen 和 Weinshall， 2019]。在多出口推理架构中，课程学习被用于在严格的延迟约束下提升早退分类器的准确率，方法是在训练过程中逐步引入更困难的样本 [Bakhtiarnia 等人， 2021]。

2.6 我们的贡献

与以往工作不同，我们在从中间 Transformer 层提取的隐藏状态特征上训练轻量级预测器，以预测在满足给定准确率阈值所需的逐查询推理长度，且在任何生成开始之前。随后，我们在固定的总令牌预算下，通过贪心算法求解全局批次分配。据我们所知，我们是首个系统分析哪些 Transformer 层能为此类预测提供最丰富信号的工作，并证明中间层在推理长度估计方面具有更高的准确率。

3 方法

本工作主要受到文献中两项近期进展的启发：推理预算的自适应分配，由Damani 等人 [2024] 提出，以及由Fu 等人 [2024a] 在其 Dynasor 系统中引入的动态调度框架。尽管Damani 等人 [2024] 侧重于调整每个查询所分配的推理轨迹数量，而Fu 等人 [2024a] 强调在固定的推理预算截止阈值下调整规模（每条轨迹的令牌预算），我们的方法则利用隐藏状态中的丰富信息，动态分配每个查询的推理预算。

3.1 数据集预处理与实验设置

我们在GSM8K 数据集上进行了实验 [Cobbe 等人， 2021], 这是一个小学数学文字题基准。该数据集包含 7,450 个训练样例和 1,294 个测试样本。每个样本包含一道自然语言数学问题及其解答，解答由一系列推理步骤构成，并以最终数值答案结尾。

3.1.1 数据处理

对于训练集和测试集中的每个问题，我们先使用 DeepSeek 分词器对问题进行分词，记录其输入长度Q，从而生成一个 16 维的提前停止概率向量。随后，我们使用温度参数 0.7 和 top-p 0.95, 为每个问题生成一百条相互独立的推理轨迹。在每条轨迹中，我们插入一个固定的探测字符串

> ”哦，我突然想到了整个问题的答案，**最终答案**\n\n\[\boxed{”

每隔 16 个令牌插入一次，最多不超过 256 个令牌；该探测字符串会迫使模型在此处给出最终答案。每次插入探测字符串后，我们都会提取模型的答案，并将其与真实值进行比较。最后，对于每个探测点（16、32、.....、256 个令牌），我们计算一百条轨迹中答案正确的比例，从而得到相应令牌预算下的提前停止概率。

3.1.2 训练数据生成

我们的预测器训练数据由两个部分组成：输入特征和目标标签。输入特征由 DeepSeek 模型

编码器生成的隐藏状态表示构成。对于每个问题，我们提取了每个 Transformer 层（第 1 至第 28 层）中 [CLS] 令牌对应的 1536 维隐藏状态，因此每个样本有 28 个特征向量。目标标签包含两组值。首先，我们记录了长度为 16 的提前停止概率向量，其中每个条目表示在依次递增的探测点（16、32、……、256 个令牌）强制作答时观察到的正确答案比例。其次，我们根据每个问题在每次查询 256 个令牌推理预算下的答题正确概率，推导出难度标签——简单、中等或困难：将答题正确概率处于最低 20 个百分位的问题标记为“困难”，处于最高 20 个百分位的问题标记为“简单”（阈值为 $p_{20} = 0.18$ 和 $p_{80} = 0.84$），其余问题标记为“中等”。GSM8K 查询的这些特征和标签构成了我们的提前停止模型和难度分类模型的训练集和测试集。

3.1.3 数据划分与验证

我们保留了原始的 GSM8K 训练/测试划分，以确保与先前工作的可比性。最终测试集包含 1,294 个样本，仅用于最终评估。处理后的数据集统计信息如下：

划分	总计	Easy	中等	Hard
训练集	7,450	1,506	4,437	1,507
Test	1,294	271	760	263

表1：预处理和难度分层后的数据集统计信息

为确保稳健性，我们验证了测试集中的难度分布与训练集中的难度分布高度一致。我们还确认，不同划分之间的平均问题长度和词汇分布保持一致。

在 GSM8K 数据上，我们展示了提前停止概率。请注意，对于较低的令牌预算，生成答案正确的概率几乎全部集中在 0 附近。对于较高的推理预算，在给定推理预算下生成答案正确的概率则集中在 1 附近。此外，有些问题即使使用最大的推理预算，几乎总是会得到错误答案——我们将其称为困难/不可能问题。对于其他问题，即使适度增加推理预算，也会带来显著的预期准确率提升。

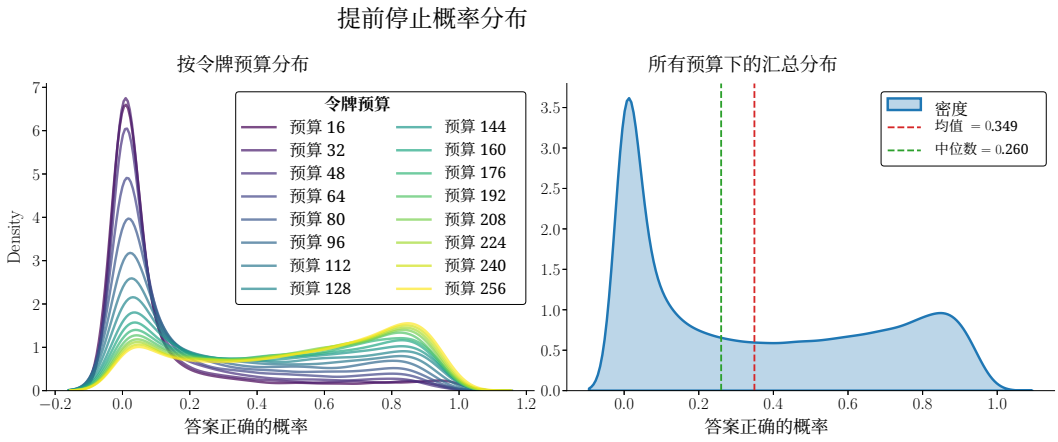

图1：(a) 各令牌预算下正确答案概率的分布；(b) 所有预算的聚合核密度估计。

3.2 提前停止预测

为困难问题分配过少的令牌会导致答案错误，而为简单问题分配过多的令牌则会浪费计算资源。通过准确预测提前停止概率——即模型在生成特定

数量的推理令牌后正确解决问题的可能性——我们可以对资源分配做出明智决策。这些预测为我们的批次分配策略奠定了基础（详见第 3.3.3节），使我们能够在不同查询之间动态分配固定的令牌预算，从而最大限度地提高总体准确率。

受到以下研究发现的启发：Damani 等人 [2024] 发现，Transformer 隐藏状态包含丰富的信号，可用于预测大语言模型在不同查询上的表现。我们据此探讨第一个研究问题：在生成之前，我们能否针对每个查询预测推理任务所需的最优令牌预算，以达到期望的准确率？

我们探索两种互补的方法，利用模型的内部表示来预测提前停止概率：(1) 在多个 Transformer 层的隐藏状态上训练轻量级多层感知机；(2) 对基础模型进行参数效率更高的 LoRA 微调。

3.2.1 基于多层感知机的早停预测器

为检验我们的假设——Transformer 隐藏状态编码了有关推理复杂度的预测信号，且某些层可能尤其有信息量——我们训练简单的多层感知机模型，将隐藏状态表示映射为提前停止概率向量。我们特别假设，Transformer 架构的中间层通常兼顾句法与语义特征，因而可能为推理难度提供最强的预测信号。

架构与训练。 我们的多层感知机预测器采用一致的架构，由两个包含 256 个隐藏单元并使用 ReLU 激活函数的全连接层，以及一个用于将输出限制在 [0,1] 的最终 Sigmoid 激活组成。输入维度对应隐藏状态大小（对于 DeepSeek-R1-Distill-Qwen-1.5B，为 1536），输出维度则与预测点数量一致（共 16 个，对应令牌预算从 16 到 256，每次递增 16）。我们针对每个 Transformer 层（1–28）分别训练多层感知机，以系统评估哪些层编码了对推理长度需求最具预测性的信号。

模型使用 Adam 优化器和均方误差（MSE）损失进行训练。我们根据验证表现进行早停，以防止过拟合。为确保每一层都能达到最佳性能，我们对隐藏层数量（1 到 3 层）、隐藏层大小（128、256 或 512 个单元）、学习率（1×10^{-4}、5×10^{-4} 或 1×10^{-3}）以及丢弃率（0.0、0.1 或 0.2）进行超参数搜索，为每一层的预测器选出最佳配置。

3.2.2 用于提前停止预测的 LoRA 微调模型

作为外部提取并处理隐藏状态的替代方案，我们假设提前停止预测本质上是一项语言任务，其中对问题陈述的语义理解与推理复杂度相关。不同于基于提取的隐藏状态进行处理的多层感知机方法，我们探索一种更集成的方法，即使用低秩适配（LoRA） [Hu 等人，2022]。我们的 LoRA 实现以原始问题文本为输入，并直接针对不同的令牌预算生成提前停止成功概率。

该方法通过在 Transformer 架构中插入可训练的低秩矩阵，同时保持预训练权重冻结，从而实现参数高效微调。

此外，这种设计利用模型预训练的语言知识来识别我们假设与推理复杂度相关的语义模式。例如，我们预计，包含“证明”“推导”或“说明”等词语的数学问题通常意味着更高的推理复杂度，需要更长的令牌预算；而包含“计算”“求出”或“评估”等指令的问题，往往意味着更直接的计算任务，可以在更早的停止点成功完成。我们的 LoRA 微调方法旨在直接从输入文本中捕捉这些语言线索，从而有望基于问题陈述的语义内容，更准确地预测提前停止。

架构与实现。 我们的实现直接在早停预测任务上对 DeepSeek-R1-Distill- Qwen-1.5B 模型进行微调。该架构由基础语言模型构成，并添加了 LoRA 适配器（秩为 $r = 16$，缩放因子为 $\alpha = 32$），作用于注意力机制中的查询和值投影矩阵。这种有针对性的方法使我们能够

高效地将模型的推理能力适配到特定任务，同时尽可能减少参数量。

预测流程。 预测流程由经过 LoRA 适配的语言模型和一个回归头组成。回归头采用两层多层感知机实现，隐藏维度为 ($h_{\text{model}}/2$)，其中 h_{model} 是基础模型的隐藏层大小（1536）。我们提取与输入序列最后一个词元对应的最终隐藏状态，并依次进行层归一化、降维至 $h_{\text{model}}/2$、ReLU 激活、线性投影至输出维度（16），以及 Sigmoid 激活，将预测值限制在 $[0, 1]$。

训练细节。 我们通过最小化预测的提前停止概率与真实值之间的均方误差损失来优化模型。训练使用 AdamW 优化器，学习率为 1×10^{-4}，共进行十个训练轮次；训练批大小为 32，评估批大小为 8。超参数调优考察了 LoRA 秩取 8、16 或 32，LoRA 缩放因子取 16、32 或 64，学习率取 5×10^{-5}、 1×10^{-4} 或 2×10^{-4}，以及回归头架构的其他备选方案。

3.3 预测问题难度以进行令牌预算分配

虽然提前停止概率向量能够对不同令牌预算下的推理成功率进行细粒度预测，但对于实际分配策略而言，这种预测可能过于详尽。我们假设，一种更简单且更易解释的方法——将问题分类为离散难度类别——能够提供有效进行令牌预算分配所需的充分信号，同时降低算法复杂度。
采用这种分类方法有双重动机：（1）离散类别可以自然地映射到分层分配策略，即为简单、中等和困难问题预先设定令牌预算；（2）与精确的概率估计相比，分类可能对噪声更具鲁棒性，尤其是在部署场景中，问题分布可能会偏离训练数据。此外，难度分类符合人们对问题复杂度的直觉，使分配决策对用户而言更易解释，也更值得信赖。

我们探索了两种互补的问题难度分类方法：一种是利用大语言模型现有能力的少样本方法，另一种是针对我们的任务专门训练的 LoRA 微调分类器。

3.3.1 使用大语言模型进行少样本分类

我们首先考察最先进的大语言模型能否在未经专门训练的情况下，有效地进行少样本难度分类。这种方法基于这样一个假设：商业上可用的模型已通过对多种语料库进行预训练，内化了大量有关数学问题复杂度的知识。

实现。 我们使用 GPT (o4-mini) 实现了一个分类流程，用于分析每道题目的陈述，并仅依据问题文本评定其难度。提示词要求模型将问题归类为简单、中等或困难，简要说明选择理由，并以适合自动化处理的结构化 JSON 格式返回结果（见附录 6.4）。

提示词通过一个简单问题的带标签示例、一个中等问题的带标签示例和一个困难问题的带标签示例来定义难度类别。
这种方法利用大语言模型中已有的数学知识和上下文学习能力，识别难度语言标记。例如，模型会识别出，包含“计算总和”或“求 x”等词语的问题通常较为简单，而涉及“证明”或需要对多个变量进行多步推理的问题则难度级别较高。JSON 格式响应便于自动化处理，并能与我们的令牌预算分配系统集成。

3.3.2 LoRA 微调分类模型

虽然少样本分类提供了一种参数高效的基线方案，但我们假设，在特定数据集上微调的专用分类模型能够获得更高的准确率。为构建这一专用分类器，我们基于 DeepSeek-R1-Distill-Qwen-1.5B 基础模型实现了一个经过 LoRA 微调的模型。

架构与实现。 该分类模型采用与早停预测器相同的输入处理方式，但输出的是离散标签，而非概率向量。基础语言模型通过秩为 $r = 16$、缩放因子为 $\alpha = 32$ 的 LoRA 适配器进行扩展，并将其应用于注意力层中的查询和值投影矩阵。分类头首先对输入序列的最终隐藏状态进行层归一化，然后将其线性投影到维度为 $h_{\text{model}}/2$ 的空间。接着应用 ReLU 激活函数，并使用概率为 0.1 的随机失活进行正则化。最后，通过线性投影映射为三个输出逻辑值，分别对应简单、中等和困难类别。

训练方法。 我们在训练过程中采用交叉熵损失，并使用准确率、精确率、召回率和 F1 分数评估性能，尤其关注模型在所有难度类别上的均衡表现。如第 3.1.2 节所述，真实难度标签根据 256 个 token 的性能数据得出，并使用训练集的第 20 和第 80 百分位数阈值（具体为 $p_{20} = 0.18$ 和 $p_{80} = 0.84$）来划分简单、中等和困难问题。

我们的实现采用批大小为 32 进行训练，批大小为 8 进行评估，使用 AdamW 优化器，学习率为 1×10^{-4}。我们最多训练模型 50 个训练轮次，并根据验证准确率进行早停，为模型学习难度模式留出充足时间，同时防止过拟合。

我们假设，与少样本方法相比，这个经过微调的分类器能够捕捉到更细致、且更贴合数学推理任务的语言模式，同时保留离散难度类别相较于连续概率向量在可解释性方面的优势。生成的分类结果可直接作为 token 预算分配策略的输入，从而根据问题难度差异化地分配资源。

3.3.3 贪心算法

训练得到的多层感知机或经过微调的 LoRA 模型在 GSM8K 训练集上训练，并在 GSM8K 测试集上进行样本外测试。
算法从每个问题至少分配 16 个令牌开始，迭代地为预测准确率增益（根据预测的解答长度估算）最高的问题分配额外的 16 令牌窗口。该过程持续进行，直至令牌预算耗尽，或预期不再有正向增益。这种方法使我们能够自适应地为预计需要较长解答的问题分配更多令牌，同时将平均令牌用量控制在指定预算范围内。

作为基线，我们考察非自适应策略的准确率（其中令牌预算在所有查询上保持一致），以及预言机策略的准确率（其中我们使用已知的真实早停正确率概率，并采用上文所用的贪心策略来分配令牌）。为使所提出的方法有用，它必须优于该非自适应基线。预言机方法提供了一种在完全知晓提前停止概率的情况下、使用所提出的贪心方法分配单查询令牌预算时所能达到的最佳准确率表现的概念。

3.4 基于难度的令牌预算分配

我们根据第 3.1.2 节所述的难度预测，将推理时查询划分为三类——简单、中等和困难。令 p_1、 p_2 和 p_3 分别表示当前推理批次中被预测为简单、中等和困难的问题所占比例。对于每个类别$k \in$ {简单、中等、困难}，定义

$$\text{acc}_k(b) = \frac{1}{N_k} \sum_{q_i \in \mathcal{C}_k} \Pr\big(\text{correct answer for } q_i \mid b \text{ reasoning tokens}\big),$$

算法 1使用预测的提前停止正确性概率进行贪心令牌分配

要求： Q，B，W，P ▷ 查询、预算、窗口大小、概率向量

确保： 使预期准确率最大化的分配

分配 $\leftarrow [W] \times |Q|$
剩余 $\leftarrow B \cdot |Q| - \sum$分配
当剩余 $\geq W$ **执行**
    收益 $\leftarrow$ COMPUTEGAINS(P，分配，W)
    如果最大值(gains) ≤ 0**则中断**
    结束如果
    $i^* \leftarrow$ 最大值参数(gains)
    分配[i^*] $+= W$
    剩余 $-= W$
结束 while
返回分配

注：*ComputeGains* 针对每个查询返回再增加一个窗口的边际预期准确率增益大小为 W（若不再有剩余窗口则为 $-\infty$）。

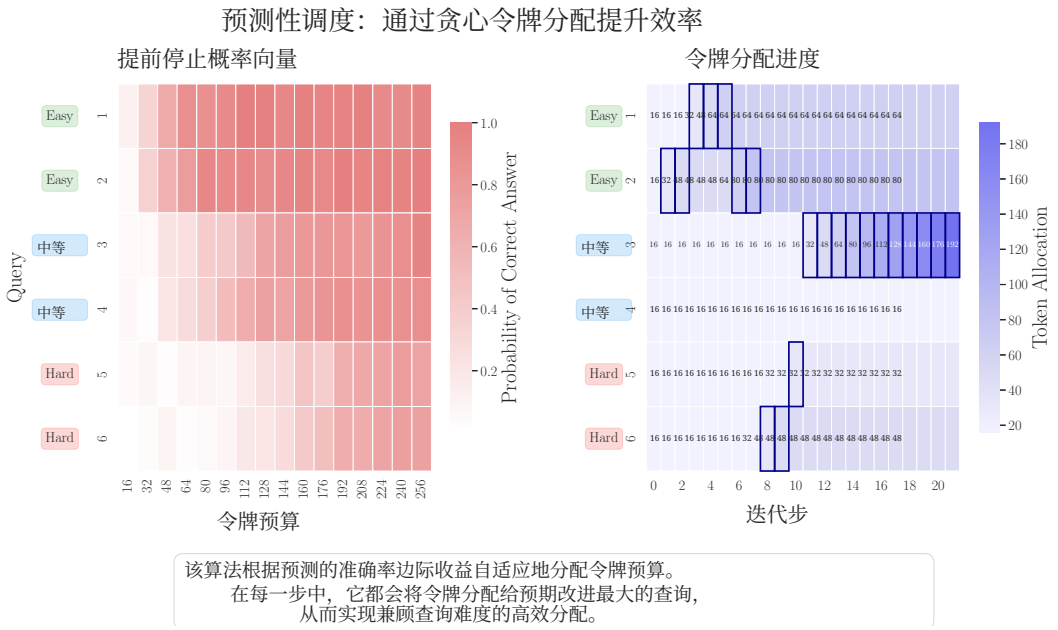

图 2：贪心令牌分配算法的可视化。**左侧：**提前停止概率向量展示了在不同令牌预算下，每个查询生成正确答案的可能性。颜色越深，表示答案正确的概率越高。**右侧：**算法执行期间在各查询间进行的渐进式令牌分配。每个单元格显示分配的令牌预算，蓝色轮廓标出每一步获得额外令牌的查询。算法首先为所有查询分配最低预算，然后迭代地将额外令牌分配给预期边际准确率增益最高的查询；起初优先处理较简单的查询，随着预算增加，再逐渐将资源转向难度较高的查询。

其中，$\mathcal{C}_k$ 是类别 k 和 $N_k = |\mathcal{C}_k|$中的所有训练样例集合。因此，$\text{acc}_k(b)$ 表示为难度为 k的问题每个查询分配 b 个令牌时的预期准确率，该值根据 GSM8K训练集 中难度为 k 的问题的所有提前停止概率向量按元素取平均计算得出。

在平均每查询预算 B 和窗口大小 W 的条件下，我们通过求解来寻找最优的每类别预算（b_1，b_2，b_3）$\in \{W, 2W, \ldots, 16W\}^3$

$$\max_{b_1,b_2,b_3} \; p_1 \,\text{acc}_{\text{easy}}(b_1) + p_2 \,\text{acc}_{\text{medium}}(b_2) + p_3 \,\text{acc}_{\text{hard}}(b_3) \quad \text{s.t.} \quad p_1 b_1 + p_2 b_2 + p_3 b_3 \leq B.$$

算法2的输入包括推理查询集合 Q、预算 B、窗口大小 W、难度预测向量以及批次比例 p_1， p_2, p_3。计算出（b_1， b_2， b_3）后，类别 k 中的每个查询都会获得 b_k 个推理令牌，如下所示。

算法2基于难度的令牌预算分配

要求： Q, B, W,难度预测 p_1, p_2, p_3
确保： 为每个查询分配预算，以最大化预期准确率
1：$(b_1, b_2, b_3) \leftarrow$ GETOPTIMALBUDGETS(B, p_1, p_2, p_3)
对于 $i = 1, \ldots, |Q|$ **执行**
    如果难度预测[i]=简单**则**
        分配[i] $\leftarrow b_1$
    否则，如果难度预测[i]=中等 **则**
        分配[i] $\leftarrow b_2$
    else
        分配[i] $\leftarrow b_3$
    结束 if
结束 for
11：**返回**分配

Note：GETOPTIMALBUDGETS 穷举搜索 $\{W, 2W, \ldots, 16W\}^3$ 中的所有三元组，并返回 e *max*化 p_1 准确$_{\text{easy}}(b_1)$ $+ p_2$ 准确$_{\text{medium}}(b_2)$ $+ p_3$准确$_{\text{hard}}(b_3)$，同时满足 $p_1 b_1 + p_2 b_2 + p_3 b_3 \leq B$。

这种分配策略结合训练集准确率曲线和推理时批次中预测的类别比例，针对每个难度类别调整推理预算，其性能优于统一预算分配（参见第 4 节）。

4 结果

4.1 早停预测结果

我们评估了两种预测提前停止概率的方法：（1）使用来自不同 Transformer 层的隐藏状态特征训练多层感知机；（2）直接处理问题文本的 LoRA 微调模型。我们的分析具体探讨了第 3.2 节提出的两个关键假设：中间 Transformer 层是否编码了更强的推理复杂度预测信号，以及问题陈述中的语言特征是否与所需的推理长度相关。

4.1.1 隐藏状态特征的逐层分析

我们的第一个假设认为，Transformer 架构中的特定层，尤其是中间层，将编码最强的推理复杂度预测信号。为检验这一假设，我们使用从 DeepSeek-R1-Distill-Qwen-1.5B 的 28 层中分别提取的隐藏状态，训练了结构相同的多层感知机；每个多层感知机都用于预测不同推理预算下答案正确的概率。
图3 提供了全面证据，支持我们的假设。中间层（尤其是第12至17层）的表现显著优于早期层和后期层，其中第16层的测试相关性最高，达到 0.742。这证实了我们的预测：中间表示在句法和语义特征之间实现了最佳平衡，而这些特征与推理难度相关。

性能分布沿模型深度呈现明显的倒 U 形，早期层（1-6）的相关性低于 0.6，而后期层（21-28）也下降到相近水平。这一模式符合我们的假设：早期层主要捕捉表层特征，不足以评估推理复杂度；而后期层则变得过于专门化，倾向于输出生成，因此无法保留通用的推理信号。
图 4 通过相关性与损失比考察预测效率，该指标衡量单位误差所对应的预测能力。这项分析进一步证实了中间层（12-17）的有效性：这些层的效率比早期层或后期层高出 15-20%。这一量化结果

有力地支持了我们的假设：中间 Transformer 层在推理复杂度预测中具有优势。

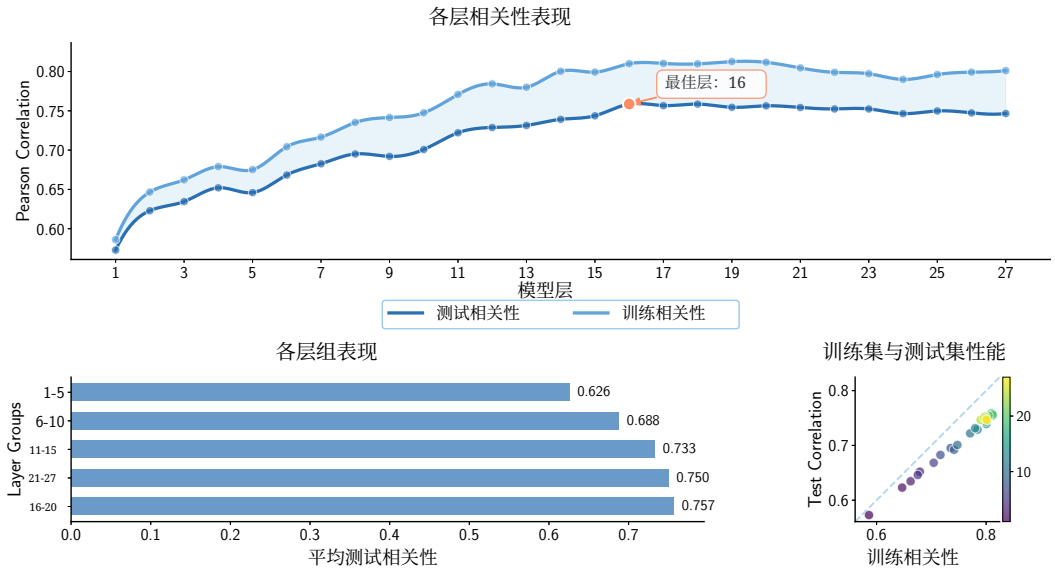

图 3：模型各层的相关性表现分析。上方面板展示了基于 DeepSeek-R1-Distill-Qwen-1.5B 模型不同层特征训练的多层感知机在测试集和训练数据集上的皮尔逊相关系数。中间层（尤其是第16层）在早停表现预测方面取得了最高相关性，这表明中间表示为推理难度预测提供了最强信号。左下方面板展示了各层组的汇总表现。右下方面板展示了训练集与测试集相关性之间的关系，数据点按层编号着色。

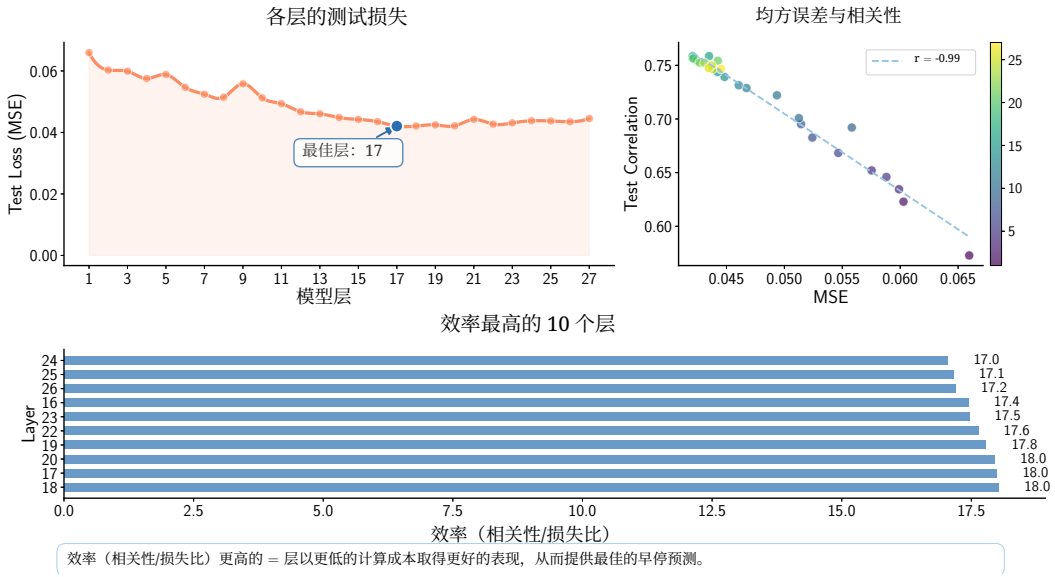

图 4：左上方面板展示了各层的测试损失（均方误差），说明预测误差如何随模型深度变化。右上方面板展示了均方误差与皮尔逊相关系数之间的关系，表明尽管这两项指标通常呈负相关，但某些层即使损失值相近，相关性仍然更高。下方面板展示了表现最佳层的相关性与损失比，指出哪些层单位损失所提供的预测价值最高。

尽管我们的结果验证了核心假设，即隐藏状态——尤其是中间层的隐藏状态——编码了有关推理复杂度的重要信息，但最高相关性仅为 0.742，这表明这种关系比最初设想的更复杂。中等程度的相关性说明，尽管 Transformer 隐藏状态包含大量预测信号，但可能还有其他未被这些表示捕捉的因素也会影响推理难度。这一部分验证表明，我们关于隐藏状态信息量的假设方向正确，但对其作用大小的估计或许过于乐观。

4.1.2 语言特征与推理复杂度

我们的第二个假设提出，问题陈述中的语言模式与推理复杂度直接相关，而经过 LoRA 微调的模型能够捕捉这些模式，从而预测提前停止概率。这种方法的评估均方误差为 0.0795，预测概率与真实概率之间的皮尔逊相关系数为 0.444。

0.444 的中等正相关表明，问题陈述中的语言特征确实包含有关推理复杂度的信号，为我们的假设提供了部分支持。然而，其相关性强度低于多层感知机方法，这表明仅凭语言特征预测精确的提前停止概率——这是一项连续且粒度细致的任务——比我们最初设想的更具挑战性。

虽然语言特征似乎不足以精确预测连续的提前停止概率向量，但它们仍可能有效预测离散难度类别（正如我们在第3.1.2节的难度分类结果中所示）。这表明，问题陈述中的语言模式可能与粗粒度难度评估的相关性高于与细粒度推理长度预测的相关性。

基于LoRA的预测器在连续提前停止预测上的表现低于我们最佳的多层感知机模型（相关性为0.444对0.742）。有几个因素可以解释这一性能差异：首先，预测16维概率向量比离散分类需要更精确的映射，这可能超出仅从语言特征中能够有效学习的范围。其次，隐藏状态可能代表已经变换过的特征，这些特征与推理表现的相关性更为直接，而原始语言特征则需要大量额外处理才能获得类似的预测能力。第三，有限范围的LoRA适配可能不足以完成从语言特征到精确提前停止概率的复杂映射，尽管它对于较粗粒度的难度分类可能已经足够。

虽然 LoRA 方法在连续概率预测方面表现较差，但仍能提供有价值的补充见解。其适度的正相关性证实，语言信号确实与推理复杂度有关，即使额外的变换处理（如中间 Transformer 层所进行的处理）能够增强这些信号。此外，该方法具有端到端的特性，因此在可能无法访问模型内部状态的部署场景中具有实际优势。

我们的结果支持了第一个假设，即中间 Transformer 层在推理复杂度预测方面具有优势。关于语言特征与推理复杂度的第二个假设则需要进一步完善——尽管语言特征确实与推理复杂度相关，但它们似乎更适合用于粗粒度难度分类，而非精确的提前停止预测。正如我们将在第 3.1.2 节中所展示的，与连续概率预测相比，同一种基于语言特征的方法用于离散难度分类时，表现要高得多。

4.1.3 利用提前停止预测进行令牌预算分配

在确认 Transformer 隐藏状态——尤其是中间层的隐藏状态——编码了有关推理复杂度的有意义信号后，我们进一步探究这些预测能否在实际部署场景中有效指导令牌预算分配。我们的目标是确定，基于提前停止预测的自适应分配策略能否在不同查询间优于均匀预算分配。

我们实现了一种贪心分配算法，利用多层感知机根据第16层（表现最佳的层）的预测结果，将固定的令牌预算分配给一批问题。该算法

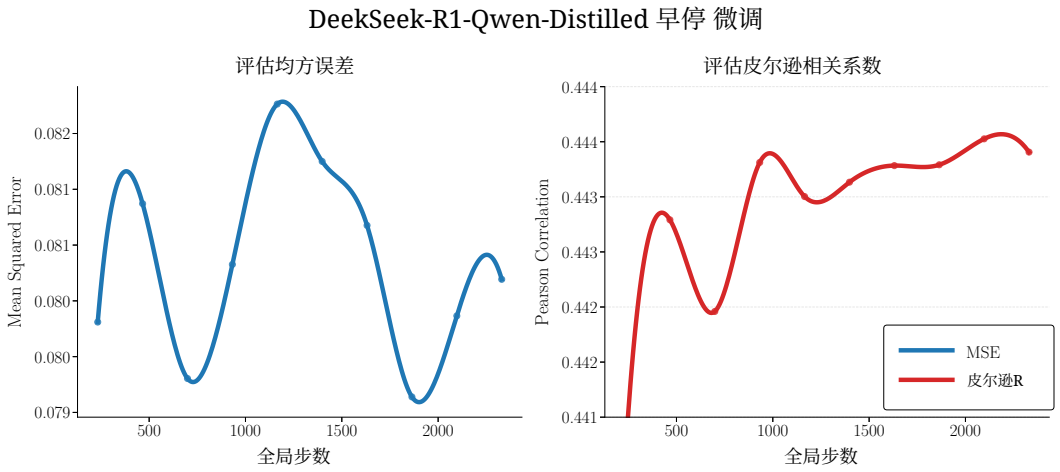

图 5：使用 LoRA 微调训练 DeepSeek-R1-Qwen-Distilled 模型进行早停预测。**左：** 评估均方误差 **右：** 预测的提前停止概率与真实提前停止概率之间的评估皮尔逊相关系数。

首先为所有问题分配最低预算，然后迭代地将额外令牌分配给那些预计能从增加推理预算中获得最大准确率提升的问题。

作为基线，我们考察非自适应策略的准确率，该策略为所有查询均匀分配令牌预算；同时也考察预言机策略的准确率，该策略使用已知的早停正确概率，并采用相同的贪心策略分配令牌。为了使我们提出的方法具有实用价值，其表现必须优于非自适应基线。预言机方法则提供了一个参照，展示在具备完美预测能力并采用贪心方法时可能达到的最佳准确率。

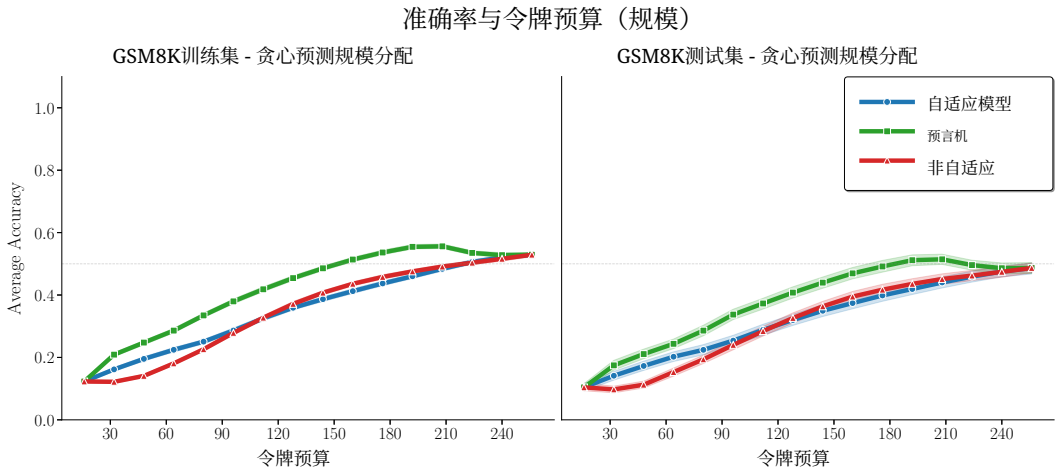

图 6：使用基于第16层隐藏状态训练的 2 层多层感知机所产生的提前停止预测进行自适应分配时，准确率与令牌预算的关系。该图比较了三种策略：(1) 基于多层感知机预测的自适应分配；(2) 采用均匀分配的非自适应基线；(3) 使用真实提前停止概率进行的预言机分配。在预算受限区间（每个查询 16–96 个令牌），自适应方法优于均匀分配；但在令牌预算较高时，其表现低于基线，这表明随着预算限制放宽，预测误差的影响会变得更加显著。

图 6 展示了这些分配策略在不同令牌预算下的对比性能。对于平均每个查询介于16-96个推理令牌的较小令牌预算，

使用多层感知机模型预测的自适应分配方法优于非自适应方法。这一改进在最为受限的预算区间（16-48个令牌）中最为显著。
然而，在平均令牌预算较高时，模型预测的准确性不足，自适应模型的表现不如非自适应基线。性能反转大约出现在每次查询 96–128 个令牌时，这表明存在一个交叉点：此时预测误差的代价开始超过自适应分配带来的收益。随着预算限制放宽，均匀分配策略最终会为大多数问题分配足够的令牌，从而降低自适应性的优势。

我们的自适应分配与预言机性能之间的差距表明，提前停止预测仍有很大的改进空间。这一差距在令牌预算较高时尤为明显，说明尽管当前的预测器能够捕捉到足够的信息，从而在受限情境下实现有效分配，但它们仍难以识别需要更长推理链的问题的最优停止点。

4.2 离散难度分类结果

在提前停止预测工作的基础上，我们研究了将问题划分为离散难度类别（简单、中等、困难）以进行令牌预算分配的有效性。本节展示以下方面的研究结果：(1) 使用平均提前停止向量进行基于难度的分类是否有效；(2) 少样本难度分类器与微调后的难度分类器的性能比较；以及 (3) 基于难度的令牌分配的有效性。

4.2.1 基于难度的分类验证

在实施基于难度的分配策略之前，我们需要验证离散分类是否能够有效反映提前停止行为的差异。具体而言，我们考察了对每个难度类别中的提前停止概率向量进行逐元素平均，是否能为分配决策提供合理依据。

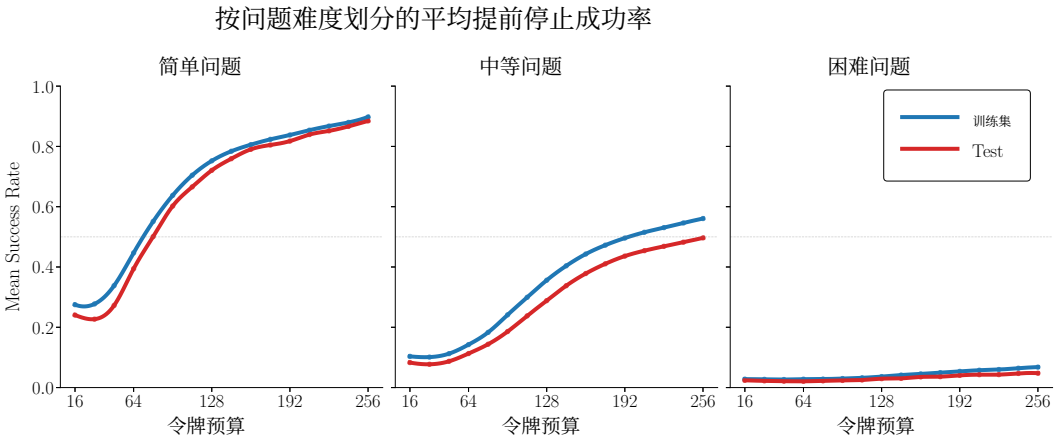

图7：各难度类别中的问题（从左至右依次为简单、中等、困难）的提前停止概率向量均值。横轴表示令牌预算，从16到256个令牌，每次递增16个令牌；纵轴表示生成正确答案的概率。蓝线表示训练集均值，红线表示测试集均值。请注意，每个难度类别中训练集和测试集的模式一致，这验证了我们的分类方法。简单问题在增加令牌后迅速改善，但较早趋于平稳；中等问题在整个令牌范围内呈现较为渐进的改善；困难问题的成功概率始终较低，直到令牌预算达到最高值时才有所提升。

图 7 展示了三个难度类别的早停概率曲线比较分析。图中呈现出明显不同的模式：简单问题（左图）随着令牌预算增加，成功概率迅速提升；中等问题（中图）的提升较为缓慢；困难问题（右图）在大多数令牌预算下的成功概率始终较低。

训练集和测试集的平均向量（分别以蓝线和红线表示）在各难度类别中走势接近，表明这些类别划分捕捉到了一致的模式，并且这些模式能够泛化到不同的数据划分。困难类别在不同令牌预算下表现得最为稳定：在大多数预算水平下，成功概率都接近于零，直到预算最高时才略有提升。相比之下，简单类别的变化幅度最大，成功概率在 16 至 96 个令牌之间急剧上升，随后趋于平稳。

本分析证实，我们的三分难度分类能够捕捉到本质上不同的推理模式，而且这些模式在不同数据划分中始终存在。训练集与测试集曲线的一致性验证了我们的方法，即以平均提前停止向量为基础，制定基于难度的令牌分配策略。

4.2.2 难度分类模型的比较分析

为检验我们关于问题陈述中的语言特征与推理复杂度相关的假设，我们实现并评估了两种难度分类方法：（1）使用 o4-mini-high 模型进行少样本分类；（2）对 DeepSeek-1.5B 模型进行 LoRA 微调。

结果为我们的语言复杂度假设提供了有力证据。LoRA 微调模型的表现显著优于少样本方法，测试准确率达到 66.3%，而少样本分类为 41.6%。这一 24.7 个百分点的提升表明，问题陈述中的语言模式包含足够的信息，可用于有效的难度分类。

图 8 展示了两种分类方法的混淆矩阵。LoRA 微调模型的对角元素更明显，表明其在所有难度级别上的分类效果更好。这一性能模式符合我们的假设：问题陈述中包含难度语言标记——例如，较简单问题中的“计算”或“求解”，以及较难问题中的“证明”或多步推理指标。值得注意的是，两种模型都最难处理中等类别，常将这些问题误分类为简单或困难。这种混淆很可能源于类别边界固有的模糊性，因为我们的难度分类将离散标签赋予了本质上是一个连续的推理复杂度谱。

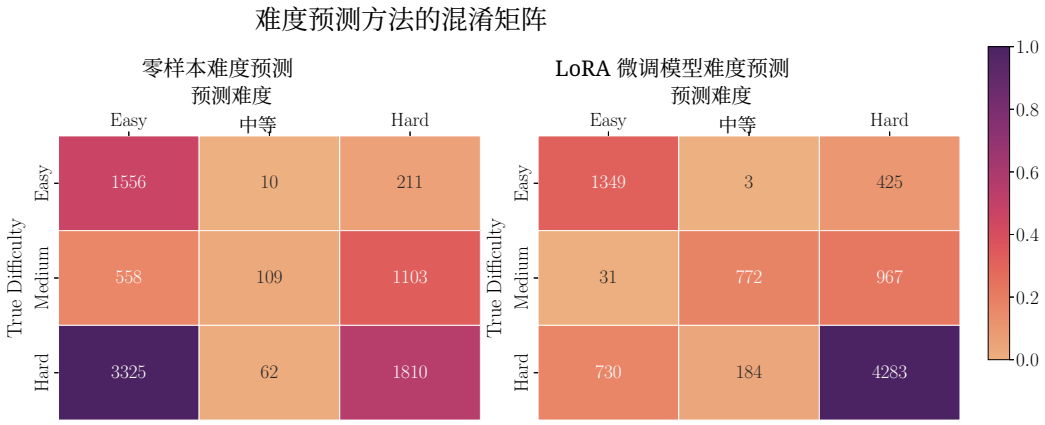

图 8：少样本方法和 LoRA 微调预测模型对难度的预测结果与实际难度的混淆矩阵。这些矩阵表明，两个模型在中等类别上的表现都最差，不过 LoRA 微调模型（右侧）的对角元素明显更强（总体准确率为 66.3%），高于少样本方法（左侧，准确率为 41.6%）。

将这些结果与我们此前关于提前停止预测的发现进行比较，可以为我们先前提出的语言复杂度假设提供重要的补充说明。语言特征不足以精确预测连续的提前停止概率向量（相关性仅为 0.444），但在离散难度分类方面表现出强得多的预测能力（准确率为 66.3%）。这种对比表明，我们最初的假设需要进一步完善：问题陈述中的语言模式确实与推理复杂度相关，但通过粗粒度分类比通过细粒度连续预测更能有效地捕捉这种关系。

尽管语言特征在详细的提前停止预测方面存在局限，但其在难度分类中的成功应用表明，我们关于推理复杂度语言标记的假设原则上是正确的，只是需要进一步明确合适的预测粒度。

4.2.3 基于难度的令牌分配性能

借助难度分类器，我们实现了一种贪心令牌分配算法，根据问题的预测难度级别，将固定的令牌预算分配给各个问题。该算法首先为每个问题分配至少 16 个令牌，然后根据各难度类别的预期准确率提升，迭代地分配额外的 16 令牌窗口。

这种分配策略依赖于根据图 7 中所示的平均提前停止向量预先计算的映射，这些映射将难度级别对应到每个令牌窗口带来的预期准确率提升。在每次迭代中，算法优先将令牌分配给预测边际准确率增益最高的问题，直至令牌预算耗尽或预计不再有正向增益。

图9 展示了少样本和基于 LoRA 的难度预测所对应的准确率—令牌预算曲线。在大多数令牌预算下，基于 LoRA 的方法优于少样本分类，并且在中等预算（每个问题约 80–160 个令牌）下，性能差距进一步扩大。

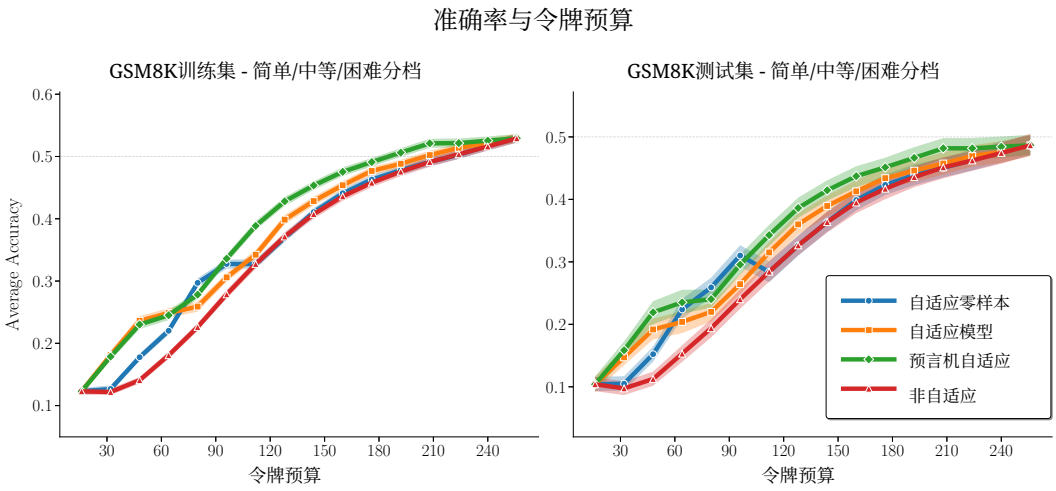

图 9：使用少样本或 LoRA 微调的难度预测进行令牌分配时的准确率与令牌预算关系图。该图表明，在所有预算水平下，LoRA 微调的难度分类始终能够实现更有效的令牌分配；在中等令牌预算（每个问题 80–160 个令牌）下，其性能优势尤为明显。

分配模式本身也揭示了最优资源分配的规律。图 10显示，在令牌预算受限时，算法会优先将资源大量分配给简单问题，因为这些问题能够带来最高的单位令牌准确率边际增益。随着预算增加，分配会逐渐转向中等问题，并在预算最高时最终转向困难问题。这种行为自然源于预期准确率曲线——简单问题在初始阶段提升迅速，但很快达到饱和；而困难问题则需要投入大量令牌后，才会出现有意义的改进。

在比较不同分配策略时，我们的基于难度的方法在中等令牌预算下尤为有效，因为此时的约束要求做出非平凡的分配决策。在这些预算区间中，基于难度的分配能够找出最能从额外计算中受益的问题，因此优于均匀分配。有趣的是，尽管基于难度的分配采用的分类比我们的连续早停预测器更粗略，但在许多预算区间中，它仍能取得相当或更好的表现。这表明，将问题简化为离散类别，既保留了有效分配所需的大部分信号，也能更好地抵御预测噪声。

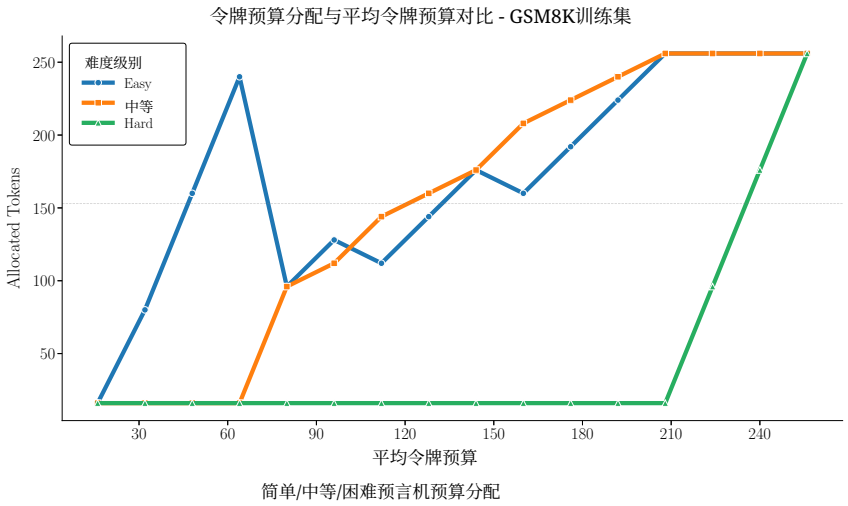

图10：在较低的令牌预算下，在预言机贪心算法中，大部分推理预算会分配给简单问题，以最大化准确率提升。随着令牌预算增加，更多令牌会分配给中等问题，最后再分配给困难问题，以最大化预期准确率提升。这种分配模式自然源于各难度类别不同的预期准确率变化轨迹：简单问题初始回报较高，但很快达到饱和；而困难问题则需要大量投入后，才能展现出显著的改进。

基于难度的分配表现出色，尤其是在使用基于 LoRA 的分类器时，这为我们修正后的语言学假设提供了有力支持。尽管从语言特征预测精确的提前停止概率颇具挑战，但依据这些相同特征将问题归入不同难度类别却十分有效。这一发现表明，问题陈述中的语言模式确实与推理复杂度相关，但与连续向量预测相比，通过离散分类能够更稳健地捕捉这种关系。

4.3 规模与难度：哪种预测框架能实现更优的分配?

分别评估提前停止预测和难度分类方法后，一个关键问题仍待解答：在实际部署场景中，哪种框架能更有效地指导令牌预算分配？为直接回答这一问题，我们在相同的令牌预算约束下，对基于这两种方法的分配策略进行了比较分析。

图11 并列展示了不同分配方法在固定单查询令牌预算下所达到的准确率。结果呈现出一致的趋势：在训练集和测试集上，基于难度的分配在所有评估的令牌预算下均优于基于规模的分配。尽管与连续的提前停止概率向量相比，三分难度分类的粒度看似更粗，这一性能优势依然存在。

有几个因素可能解释了基于难度的分配为何出乎意料地有效。首先，离散分类能够抵御预测噪声：提前停止概率估计中的微小误差可能会显著影响分配决策，而相邻难度类别之间的误分类通常只会导致轻微的分配调整。其次，基于难度的方法能够捕捉连续的提前停止曲线未能充分体现的、定性上不同的推理模式，例如在特定令牌阈值处准确率突然跃升的问题。第三，基于难度类别的分配算法可以利用更稳定的类别级统计数据来估计预期准确率提升，从而降低受离群模式影响的风险。

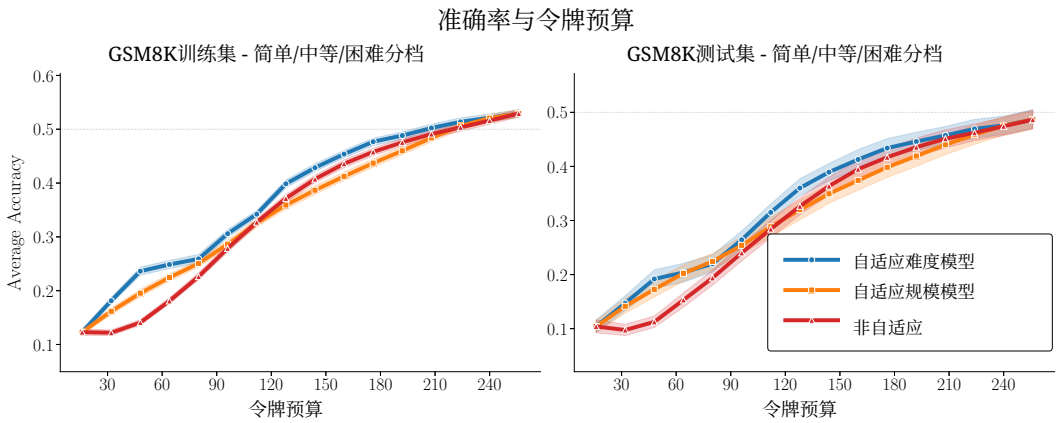

图 11：不同固定单查询令牌预算下分配策略的比较分析。该图展示了不同分配方法所达到的准确率：基于难度的分配（使用 LoRA 微调预测）、基于规模的分配（使用多层感知机第16层的预测）以及非自适应基线（均匀分配）。结果表明，在所有评估的令牌预算下，根据难度预测动态分配资源，在训练集和测试集上的表现始终优于基于规模的分配；此外，在大多数预算区间内，两种自适应方法均优于非自适应基线。

5 结论

我们提出了一种面向大语言模型推理时推理的框架，利用快速的预运行预测器来估计每个查询的最佳推理轨迹长度或难度类别。我们的系统性研究揭示了若干重要发现：(1) Transformer 层的隐藏状态特征包含大量关于推理复杂度的预测信息，其中中间层（12–17 层）提供的信号最强；(2) 问题陈述中的语言特征与推理难度相关，但与连续的提前停止概率预测相比，通过离散难度分类能够更有效地捕捉这种关系；(3) 在所有令牌预算下，粗粒度的基于难度的分配始终优于细粒度基于规模的分配。

在 GSM8K基准上的实证评估表明，我们的预测性调度方法在相同成本下，相较于均匀令牌预算分配，准确率最高可提升7.9个百分点。值得注意的是，采用 LoRA 微调分类的基于难度的分配策略效果最佳，这表明稳健的类别级预测比精确但可能带有噪声的连续估计能为资源分配提供更可靠的指导。最优分配模式会随令牌预算约束而变化：预算较低时优先处理简单问题，随着约束逐步放宽，再逐渐将资源转向中等和困难问题。这凸显了自适应策略相较于固定分配方案的价值。

展望未来，我们看到几个颇具前景的方向：（1）研究混合方法，结合隐藏状态特征与语言模式识别各自的优势；（2）将预测性调度扩展至多轨迹聚合方法，例如自洽性或思维树；（3）整合不确定性估计，以防止资源分配不当，尤其是在令牌预算较高、预测误差影响更为显著的情况下；（4）在延迟敏感、成本受限的生产系统中部署我们的方法。通过使计算资源分配与单查询复杂度相匹配，预测性调度为实现可扩展、成本高效且准确率高的大语言模型推理提供了一条切实可行的途径，而且无需修改底层语言模型架构。

参考文献

Arian Bakhtiarnia、Qi Zhang 和 Alexandros Iosifidis。通过课程学习提高多出口架构中早期退出的准确率。载于第*29*届*ACM*国际多媒体会议论文集，第10–16页，2021年。

Yoshua Bengio、J´erˆome Louradour、Ronan Collobert 和 Jason Weston。课程学习。载于第*26*届国际机器学习会议论文集，第41–48页，2009年。Karl Cobbe、Vineet Kosaraju、Mohammad Bavarian、Mark Chen、Heewoo Jun、Lukasz Kaiser、Matthias Plappert、Jerry Tworek、Jacob Hilton、Reiichiro Nakano、Christopher Hesse 和 John Schulman。训练验证器以解决数学文字题，2021年。网址：https://arxiv.org/abs/2110.14168。Mehul Damani、Idan Shenfeld、Andi Peng、Andreea Bobu 和 Jacob Andreas。学习如何进行深度思考：语言模型计算的输入自适应分配，2024年。网址：https://arxiv.org/abs/2410.04707。M Elbayad 等。深度自适应 Transformer。载于第*37*届国际机器学习会议论文集，2020年。

William Fedus 等。Switch Transformer：通过简单高效的稀疏性扩展至万亿参数模型。载于第*38*届国际机器学习会议论文集，2021年。

Yichao Fu、Junda Chen、Siqi Zhu、Zheyu Fu、Zhongdongming Dai、AurickQiao 和 HaoZhang。使用 certaindex 高效服务大语言模型推理程序。*arXiv* 预印本 *arXiv:2412.20993*，2024年a。

Yichao Fu、Siqi Zhu、Runlong Su、Aurick Qiao、Ion Stoica 和 Hao Zhang。通过学习排序实现高效的大语言模型调度，2024年b。网址https://arxiv.org/abs/2408.15792。

Guy Hacohen 和 Daphna Weinshall。课程学习在训练深度网络中的作用。载于第*36*届国际机器学习会议，第2535–2543页，2019年。

Tingxu Han、Chunrong Fang、Shiyu Zhao、Shiqing Ma、Zhenyu Chen 和 Zhenting Wang。《Token-budget-aware 大语言模型推理。*arXiv* 预印本 *arXiv:2412.18547*，2024年。

Edward J Hu、Yelong Shen、Phillip Wallis、Zeyuan Allen-Zhu、Yuanzhi Li、Shean Wang、Lu Wang、Weizhu Chen 等。《大语言模型的 LoRA：低秩适配》。*ICLR*，1(2):3，2022年。

Michael P. Kumar、Benjamin Packer 和 Daphne Koller。《潜变量模型的自定进度学习》。载于神经信息处理系统进展，第23卷，第1189–1197页，2010年。

Yiwei Li、Peiwen Yuan、Shaoxiong Feng、Boyuan Pan、Xinglin Wang、Bin Sun、Heda Wang 和 Kan Li。《摆脱高昂成本：面向多步推理的提前停止的自一致性》，2024年。网址 https://arxiv.org/abs/2401.10480。

Chuan Liu 等。vLLM：大语言模型的快速推理。载于 *2023*年自然语言处理实证方法会议论文集，2023年。

Xinyi Wang 等。自洽性提升语言模型中的思维链推理。载于神经信息处理系统进展，2022年。

Jason Wei 等。思维链提示能够引出大语言模型的推理能力。*arXiv* 预印本 *arXiv:2201.11903*，2022年。

Yuyang Wu、Yifei Wang、Tianqi Du、Stefanie Jegelka 和 Yisen Wang。少即是多：理解大语言模型中的思维链长度。*arXiv* 预印本 *arXiv:2502.07266*，2025年。

Ziyu Yao 等。思维树：借助大语言模型进行审慎的问题求解。*arXiv* 预印本 *arXiv:2303.11171*，2023年。

Yifan Zhang 等人. Transformer模型的高效推理调度. 收录于第*38*届国际机器学习会议论文集，2021年。

6 附录

6.1 答案正确性评估

对于 GSM8K 中的每个问题，我们都使用严格的提取和比较流程，将模型的预测答案与真实值进行比较。首先，通过定位最后一次出现的 \boxed{...} 中的文本，确定最终数值答案。接下来，去除除小数点以外的所有非数字字符。然后将清理后的字符串转换为浮点数值，并在相对容差为 10^{-4} 的条件下与真实值答案进行比较，以容许细微的数值差异。如果在任何步骤中提取或转换失败——例如，未找到 \boxed{...} 格式——则将答案标记为错误。这样可以确保只有完全正确的数值输出才算成功。

6.2 多层感知机架构与超参数搜索

我们的多层感知机预测器采用一层或两层隐藏层。对于单层变体，隐藏维度设为 128、256 或 512 个单元；对于双层变体，两层均设为 128 个单元或 256 个单元。我们从对数均匀分布中采样学习率，范围为 $[10^{-3}, 10^{-2}]$；从 $[0, 0.5]$ 中均匀选取丢弃率，并将批大小设为 8、16 或 32。每种配置都在留出的验证集划分上进行评估，以确定性能最佳的架构和训练设置。

6.3 运行时与硬件

所有微调实验均在 1 个 H100 GPU 上进行，少样本难度分类所需的 API调用费用为 $< \$0.10$。

6.4 问题难度提示的少样本分类

以下提示词用于让 o4-mini-high 将问题分类为简单、中等或困难。

o4-mini-high 难度分类提示

你是一名 AI 助手，负责将数学问题分类为简单、中等或困难。我们会给你一些样本，然后要求你对一个新问题进行分类。以下是一些带有分类结果的样本问题：

<样本>
"<|User|>珍妮特的鸭子每天产 16 个蛋。她每天早上吃 3 个当早餐，每天还用 4 个给朋友烤松饼。她每天把剩下的蛋拿到农贸市场出售，每个新鲜鸭蛋卖 2 美元。她每天在农贸市场能赚多少美元？<|助手|>", 中等

"<|用户|>Arobe 使用 2 捆蓝色纤维和数量为其一半的白色纤维。总共需要多少捆？<|助手|>", 简单 "<|用户|>詹姆斯决定每周跑步 3 次，每次跑 3 个冲刺。他每次冲刺跑 60 米。他一周总共跑多少米？<|助手|>", 困难</样本>

你的任务是根据新问题与所提供样本的相似程度，将以下新问题归类为简单、中等或困难。假设你的先验判断是：20% 的问题属于简单，60% 的问题属于中等，20% 的问题属于困难。请注意不要低估待分类问题的难度——如果有理由认为它属于困难，就将其归类为困难；如果有理由认为它属于中等，就将其归类为中等。如果你想将问题归为中等而非困难，就必须有非常充分的理由，说明它为何属于中等而非困难。以下是要分类的新问题：<新问题>

{在此插入问题}
</新问题>
请考虑问题的复杂程度、解决问题所需的步骤数量，以及它与所提供样本的比较情况。然后，按照以下 JSON 格式给出类别和推理：

<输出>
{"推理": "<你对该类别的推理>" "类别": "< {简单、中等、困难}>", }</输出>

请确保你的推理清晰简洁，说明你为何根据问题的特点及其与所提供样本的比较结果选择了特定类别。