智能技术实训报告

李舒怀[1]，杨昊轩[2]，王彦琪[3]

[1]李舒怀（1；22122821；贡献比：40%；负责[编码、方法探索与设计、论文撰写]）
[2]杨昊轩（2；22122819；贡献比：30%；负责[论文撰写、数据集扩充、漏洞查找]）
[3]王彦琪（3；22122820；贡献比：30%；负责[数据处理、方法探索与实践、论文撰写]）

方案完整性 35	方案新颖性 10	汇报表现 15	报告质量 15	成绩
教师评语				

摘要:

本次实训的目的是对 Llama3-8B-Instruct 大模型进行微调使其在 2024 Amazon KDD cup 中的购物知识推理(shopping-knowledge-reasoning)赛道提供的评测指标上表现更佳，使模型更加擅长推理。我们使用的方法是用 LoRA 将初始权重矩阵进行分解，训练分解完的参数矩阵，再将其和原始参数矩阵合并，一起嵌入到到模型中供其进行评测。本次实训最终的结果是完成了对于 Llama3-8B-Instruct 大模型的微调，并且在购物知识推理任务上得到一个更好并且较高的得分。本次实训中的局限之处在于所采用的 Llama3-8B-Instruct 大模型是已经预训练完成的，若要再次训练则需要耗费大量的资源和时间，这是我们所达不到的，因此只能微调。遂提出 Mygo 架构，随后借助提示词工程、数据集增强等方式提高准确度。

本文所有代码将在课程结束后开源。[i]

关键词: 购物知识推理；大模型微调

1 引 言

随着人工智能和机器学习技术的快速发展，大规模预训练语言模型在自然语言处理（NLP）领域取得了显著的进展。这些模型通过在大规模文本数据上进行自监督学习，掌握了广泛的语言知识和复杂的语言模式。然而，尽管这些模型在许多 NLP 任务上表现出色，但它们在特定任务上的表现仍然受到限制，需要进一步的微调以提升其执行效果 。

在此背景下，Llama-3-8B-Instruct 模型应运而生。该模型的设计理念是将预训练与指令微调相结合，从而增强模型在特定任务上的执行能力。这种设计方法不仅使模型能够理解和生成自然语言，还能根据特定的指令执行任务。通过在预训练阶段融合指令数据，Llama-3-8B-Instruct 不仅具备通用的语言理解能力，还能够更加精准地执行自然语言指令任务 。

本文的主要目的是对 Llama-3-8B-Instruct 大模型进行微调，使其在 2024 Amazon KDD Cup 中的购物知识推理赛道上的表现更加出色。通过使用低秩适应（LoRA）技术对模型进行微调，我们期望在模型的推理能力和任务执行效率上取得显著的提升 。

总的来说，本报告旨在探讨 Llama-3-8B-Instruct 模型的设计原理、

技术架构及其在购物知识推理任务上的应用，并通过实验验证其有效性和优越性。

2 解决思路

2.1 购物知识推理任务介绍

购物知识推理任务旨在让大模型具有理解在线购物领域隐性知识的能力，并且让其能够通过已有知识进行各种类型的推理。大模型类似的推理能力对客户在线购物时浏览的行为有着非常大的作用，例如通过对用户的浏览行为所包含的隐性知识推理，可以衍生出用户潜在的浏览倾向。

本文希望解决的购物知识推理任务分为以下几个子任务：

1. 数字推理：购物中经常涉及计算，例如计算一包商品的总重量。该任务旨在让模型能够从上下文信息中提取数字相关信息并且进行数学计算推理。

2. 常识推理：为了让模型能够对用户进行个性化推荐，需要让模型具有很强的常识推理能力，在线上购物的交易中，日常用品占据了很大的比重，因此模型需要具有一定的常识。

3. 隐式多跳推理：这项子技能要求模型理解产品中隐式的、特定领域的知识，并推断购物实体之间的多跳关系。例如，要回答下述问题：

Question:
Given product type golf club, which of the following product categories best complement the given product type?
1. Sonar fathometer
2. Golf bag
3. Fitness band
4. Buoyancy device

Answer: 2. golf bag

模型应该执行以下推理：

(1) 理解问题：需要找到与高尔夫球杆互补的产品类别。

(2) 分析选项：选项 1：声纳测深仪与高尔夫无直接关联；选项 2：高尔夫球袋用于携带和保护高尔夫球杆，互补性强；选项 3：健身手环虽然与运动相关，但与高尔夫球杆互补性较弱；选项 4：浮力装置与高尔夫球杆无直接关联。

(3) 选择最优选项：选项 2 高尔夫球袋是携带和存放高尔夫球杆的必需品，互补性最强。

2.2 Llama3-8B-Instruct 模型

Llama3-8B-Instruct 模型的设计理念是基于将预训练和指令微调相结合，以提升模型在特定任务上的执行效果。Llama3-8B-Instruct 通过在预训练阶段融入指令数据，使得模型不仅能理解通用语言，还能更好地执行基于自然语言指令的任务。

Llama3-8B-Instruct 是基于 Transformer 架构构建的。Transformer 的多头自注意力机制使得模型能够高效地捕捉句子中各个词之间的关系，从而在理解和生成任务中表现出色。Llama3-8B-Instruct 拥有 80 亿个参数，处于大型模型的中等规模段。虽然模型参数较大，但通过优化架构和引入低秩适应技术，使得模型在保持高性能的同时，计算资源需求相对较低。

在训练过程中，Llama3-8B-Instruct 采用了多任务学习策略。这意味着模型不仅在一个任务上进行训练，而是同时在多个相关任务上学习。这种方法不仅可以提升模型在各个单独任务上的表现，还能通过任务间的知识共享，提高模型的泛化能力。

Llama3-8B-Instruct 的预训练数据集涵盖了多种语言和领域，确保模型能够学习广泛的语言知识和上下文理解能力。在指令微调阶段，选择了包含明确指令和对应任务的高质量数据集，如自然语言问答数据集、摘要生成数据集和对话系统数据集等。在预

训练阶段，Llama3-8B-Instruct 首先通过自监督学习在大规模文本数据上进行训练。自监督学习的目标是使模型能够通过上下文预测缺失的词语或短语，从而掌握语言的内在结构和规律。在此基础上，Llama3-8B-instruct 模型经过了专门的微调，以优化其在指令完成和对话场景中的表现。微调过程中，模型采用了监督微调和基于人类反馈的强化学习相结合的方法。强化学习阶段通过拒绝采样和近端策略优化进行多轮次迭代训练，以确保模型能够在多样化的对话场景中保持一致性和连贯性[1]。

Llama3-8B-Instruct 有着高效的架构设计和强大的任务执行能力，使其在文本生成、问答系统、翻译和对话系统以及文本分类等多个应用场景中具有广泛的应用前景。

Meta-Llama-3 模型共有三个规格：7B、8B、70B。鉴于本任务的实验条件，本文选择了 8B 大小的模型进行实验。经过实验，本文得出 8B 模型在保证基本常识、理解和推理能力的同时，能够最佳适配现有的硬件条件。这一模型的参数量既能提供足够的性能，又不至于超出硬件的承受能力。

在处理 2.1 中提到的任务中，本文采用如图 1 所示的框架。vLLM 是一个快速且易于使用的 LLM 推理和服务库。其与 Llama3-8B-Instruct 的关系如图 2 所示。

2.3 Lora 微调

大模型微调（fine-tuning）是指在一个已经预训练好的大型神经网络模型上，进一步训练模型以适应特定任务的过程。大模型所涉及的参数往往都有着上百亿千亿级的参数。在对预训练后的模型微调时分别有两种选择，一种是全参数微调，另一种是部分参数微调。对于全参数微调，需要加载模型所有的参数进行训练调整，能够得到较好的效果，但这个过程对硬件

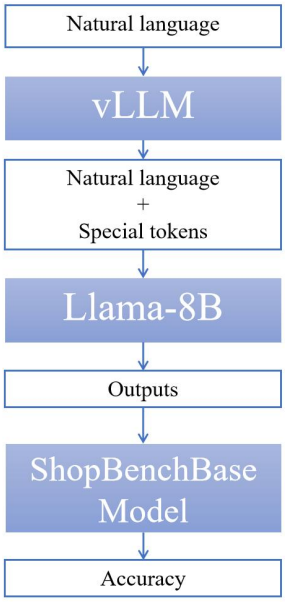


图 1 任务流程图

图 2 vLLM 示意图

依赖大并且需要耗费大量的资源时间。对于部分参数的微调：这个过程只抽取出神经网络中部分层的权重对其进行训练调整，能够减少存储空间和加速部署，但存在一些性能和模型质量的损耗。

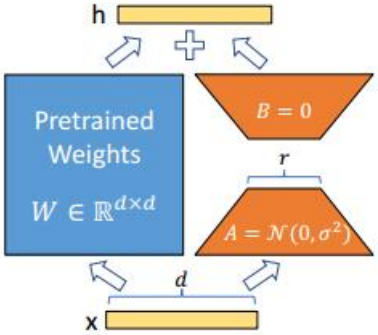
由于 Llama3-8B-Instruct 模型的参数量较大，训练所有参数的难度显著增加，所需的显存容量也呈指数级增长。因此，全面训练整个模型在硬件资源有限的情况下并不现实。为了解决这一问题，我们采用了 Lora 微调技术。

Lora 在保留基线模型所有参数的同时冻结了预训练的模型权重，并将可训练的秩分解矩阵注入到 Transformer 架构的每一层，从而大大

减少了下游任务的可训练参数的数量[2]。Lora 是一种高效的微调技术，它可以大大减少微调过程对硬件的依赖加速微调进程。

神经网络的密集层通常执行矩阵乘法，其权重矩阵是全秩的。研究表明，预训练的语言模型即使在较小的子空间中，也能高效学习，因为它们具有较低的内在维度。这表明模型在较低维度上也能很好地适应任务。

图 3 Lora 原理示意图[2]



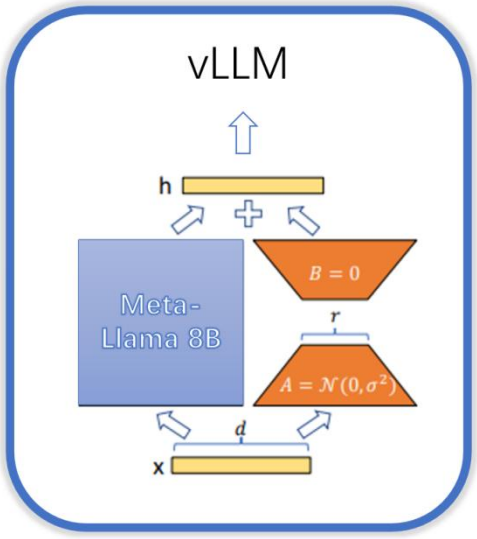
如图 3 所示，Lora 对于预训练好的基线模型参数 X 保持不变，将其分解为 A 和 B 两个低秩矩阵，仅对 A 和 B 进行训练微调，最终合并得到新的权重矩阵 h。在训练过程中，W_0 被冻结，不接受梯度更新，而 A 和 B 包含可训练参数。参数迭代传播过程数学表达式如下：

$$h = W_0 x + \Delta W_x = W_0 x + BAx \quad (1)$$

表达式(1)中 A 矩阵维度为 r * k，B 矩阵维度为 d * r，r << min(d, k)。这样处理的好处非常明显，假设矩阵 W 的维度为 1000×1000，分解后的低秩矩阵的秩为 8，不使用 Lora 时需要对 1000×1000 规模的参数进行计算，而使用 Lora 后只需要对 1000×8+8×1000 共计 16000 参数进行训练调优，大大加快了微调的进程。

将微调技术与上述大模型相结合，并融合入本任务已构建的架构后，得到了 **Mygo**（**My** purchase is your **go**al）模型架构。详见图 4。

图 4 Mygo 架构



在此架构之上，我们后续将通过数据集增强和以思维链为代表的提示词工程帮助该架构在KDD任务上获得更好的表现。

2.4 数据集

本文使用的数据集分为训练集和测试集，均为 json 格式，包含以下字段：

1. input_field：该字段包含指令和模型应该回答的问题。
2. output_field：该字段包含问题的真实答案。
3. task_name：该字段包含任务编号。
4. task_type：该字段包含任务类型。
5. metric：该字段包含用于评估问题的指标。
6. is_multiple_choice：该字段包括问题是否是多选题。
7. track：该字段包含 KDD cup 的赛道。

对于购物知识推理任务，本文的数据中 task_name 对应 task8-task10。不同 task 对应的任务解释如表 1 所示。

表 1 不同任务的解释

	Explanation
Task8	数字推理
Task9	常识推理
Task10	隐式多跳推理

本文的数据来源分别来自人工筛选后的 ChatGPT 生成数据以及来自 Amazon 官网的真实数据。我们通过给 ChatGPT 相应的提示词，生成了大量购物知识推理的相关数据，在通过人为筛选后保证了数据的质量。对于一些特定类型的数据比如大模型评估准确度较低的数据，本文通过爬虫等手段，人工构建了仅百条数据，以保证数据的真实性。

本文通过增加数据集数量和提升数据集质量的方式，将数据堆积以适应 Lora 微调训练大模型。

2.5 思维链

本文同时关注到能增强大模型推理能力的诸多技术中的提示词工程。鉴于本任务中逻辑推理和数学计算的内容占比不少，我们关注思维链技术。思维链技术在数学和逻辑推理等需要多步推理的问题上特别有效。它通过将复杂问题分解为简单步骤，引导模型逐步推理，从而提高了问题解决的准确性和效率。这种方法不仅降低了推理过程的复杂性，还增强了模型的解释性和透明性，是提升大模型推理能力的重要工具。以下是对该技术的具体介绍。

思维链技术（Chain-of-Thought, CoT）是提示词工程中一种增强大模型推理能力的方法[3]。它通过引导模型逐步进行多步推理，从而解决复杂任务。这种技术的核心理念是让模型在推理过程中逐步显式地记录每一个中间步骤，类似于人类在思考问题时的思路展开。以下是思维链技术的详细介绍：

1. 逐步推理：思维链技术通过将复杂问题分解为一系列简单的步骤，引导模型逐步推理。每一步的输出作为下一步的输入，从而完成整个推理过程。

2. 提示词设计：为了实现逐步推理，需要精心设计提示词。这些提示词不仅包含问题本身，还包含引导模型进行逐步思考的指示。例如，提示词可以包括“首先…其次…然后…”等，引导模型按步骤进行推理。

3. 中间步骤显式表达：在推理过程中，模型需要显式地表达每一个中间步骤的结果。这不仅有助于追踪模型的思维过程，还能提高最终答案的准确性。

我们尝试利用这一提示词技术增强模型微调后在推理和数学计算等问题上的回答准确率。

3 实验及结果分析

3.1 基准实验

根据 KDD cup 提供的测试代码，本文进行了初步的实验并获得了基准结果。这些结果将作为检验后续改进效果的基准。在随后的实验中所使用的环境和硬件条件与初步实验保持一致。通过这种方式，能够精确衡量模型性能的提升和进步。

在初步实验中，本文配置了实验环境和硬件条件，以确保结果的准确性和可重复性。这些配置包括操作系统版本、依赖库版本、硬件规格（如 CPU、GPU、内存）以及其他相关参数。在后续的改进实验中，本文严格遵循这些配置，以确保每次实验的结果具有可比性。硬件条件如表 2 所示：

表 2：硬件设备

硬件设备	参数
GPU	NVIDIA-3090
GPU 数量	1
GPU 显存	24 GB
CPU	Intel(R)Xeon(R)CPU E5-2683 v4
内存	62G
显卡驱动版本	530.30.02
CUDA 版本	12.1

关于环境配置，可以参见仓库。

通过保持环境和硬件条件的一致性，能够排除外部变量的影响，专注于模型自身的改进效果。这样一来，

每次优化所带来的实际性能提升可以得到更精确地评估，确保本文中改进措施的有效性和可靠性。

依据上述环境与硬件条件，基准实验结果如表 3 所示：

表 3：基准实验结果

任务编号	准确度
Task8	0.25
Task9	0.50
Task10	1.00
平均	0.583

3.2 微调结果

依据前述的 Meta-Llama-3 的大模型 Lora 微调技术，本文尝试对大模型进行微调。其中，Lora 的配置用于规定训练的参数量，即待训练的矩阵；训练参数即规定训练的轮次、小批量的大小、学习率等超参数。参数列表如表 4、表 5 所示：

表 4：Lora 配置

超参数	设定内容
任务种类	CAUSAL_LM
秩	4
Lora alaph	64
Lora dropout	0.1

表 5：训练超参数设置

超参数	设定内容
批量大小	16
累积步数	4
训练轮数	4
学习率	1e-4

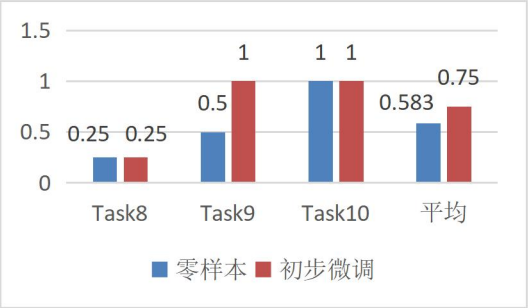
本文将训练过后的 Lora 参数借用 vLLM 的 API 送入 Meta-Llama-3-8B 的模型中，完成 Lora 微调结果和初始大模型的参数的融合，以期完成迁移学习。在该下游任务上得到了相比基准更优秀的结果。结果列表如表 6 所示：

表 6：微调后结果

任务名称	准确率
Task8	0.25
Task9	1.00
Task10	1.00
平均	0.75

可见，微调后对于 task9 有明显的提升。在其他任务上没有表现更优。具体的结果对比如下：

表 7：零样本与初步微调的对比



3.3 思维链结果

通过 3.2 节的实验结果可以发现，task8 的准确度较低，该任务包含算数问题和推理问题。借助上述的思维链为主的提示词工程技术，通过在提示词阶段人工加入相关引导，以期对其进行改进。

提示词加入的内容具体如下：

Prompt:
You are a helpful online shopping assistant. Please answer the following question about online shopping and follow the given instructions.
Let's think step by step.

除此之外，本文还尝试将理由加入了测试集中，修改后的数据内容示例如下：

{"input_field":"The product 'Acer Nitro 5 Gaming Laptop, 15.6\" FHD, Intel Core i5, 8GB RAM, 256GB SSD, NVIDIA GeForce GTX 1650, Windows 10 Home' appears on an e-commerce website. It is a gaming laptop. How many USB ports are there?\n0. It cannot be inferred.\n1. 1\n2. 2\n3. 3\nAnswer: ","output_field": 3,"explain":"Gaming laptops like the Acer Nitro 5 typically have three USB ports for connectivity.","task_name":"task8","task_type":"multiple-choice","metric":"accuracy","is_multiple_choice":

true,"track":"amazon-kdd-cup-24-shopping-knowledge-reasoning"}

表 7：增加思维链后结果

任务名称	准确率
Task8	0.25
Task9	1.00
Task10	1.00
平均	0.75

进行上述处理后，实验得到加入思维链和推理理由的结果如表 7 所示。可见思维链对结果提升并不明显。

3.4 针对性的增加数据

将错误的题目在经过模型评测之后打印出来，实验发现了模型对于 task8 中的算术题以及推理题无法达到较好的准确率。

本文有针对性地对这两类问题生成数据。对于算数题，本文使用 ChatGPT 等大模型批量生成。通过输入数据格式样例，以及适当的提示词，我们得到了近 500 条算术题数据。对于推理题：若使用 ChatGPT 等大模型则会出现大量答案错误的数据，本文选择使用人工构建。

本文通过使用爬虫等手段在 Amazon 官网上搜寻合适的商品数据，并将其按照数据样例进行格式化，最终得到了近 100 条推理题数据。表 8 是结果展示，其中 task8, task9, task10 测评数据分别为 8 条，4 条，4 条。从表 8 中可以看到在针对性的增加数据集之后 task8 的准确率从 0.25 提升到了 0.75。

表 8 数据增强后的评测结果

3.5 未能解决的问题

由于原有测试集的数量过少，可能导致模型评估结果不够准确或具有偶然性，因此本文尝试将评测数据增加到至 66 条以此来增加模型评测的可信度，并最终得到了表 9 所示结果，其中 task8, task9, task10 测评数据分别为 23 条，19 条，24 条。

表 9 增加测试集数量后的结果

由表 9 可以看到在加入更多的测试集后模型的总体准确度有了巨大的提升，这很好的排除了模型的偶然性。然而，模型的错题与 3.4 节中所提到的错题一致。为此本文尝试不同的解决方法例如增加数据量、思维链的应用(增加提示词)以及在每条数据中新开一个字段“explain”，然而最终效果不加，甚至出现了准确率逐步降低的情况。通过调整 Lora 参数后，本文排除了“因为模型过拟合而导致模型泛化能力降低使其无法正确做出回答”的可能性。

实验还尝试对模型进行专门微调，即对 task8 微调专门的 Lora，对 task9, task10 使用另一个微调完的 Lora。但是由于在调取时出现了显存不够的情况。在一次代码执行过程中仅支持一个模型参数的加载。因此我们只尝试了在 task8 的计算子任务和推理子任务上分别试验不同的微调模型。

3.6 最终结果

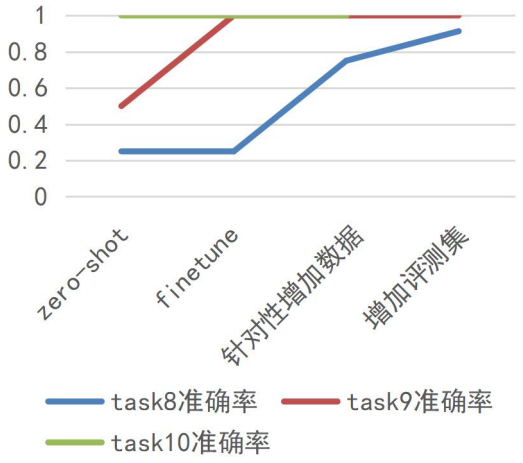
在经过微调、数据集增强、测评集增强后，本文得到的最佳结果如下：

表 10：最终结果

任务名称	准确率
Task8	0.913
Task9	1.000
Task10	0.958
平均	0.957

尽管本文在最终的评测结果上得到更好且较高的分数，但本文最后未能将 task8 准确率提升到百分之百。图 4 展示的是整个实训过程中 task8, task9, task10 准确率的变化图。

图 5　task8, 9, 10准确率变化图



4　总结

4.1 实训总结

本次实训要求深入学习并掌握大模型微调的概念和方法。通过对 Lora 的应用，小组成功地在计算资源以及显存资源有限的情况下对 Llama3-8B-Instruct 大模型进行了微调，并取得了比基准测试优异许多的成绩。

小组成员随后尝试了包括提示词工程、人工数据标注、数据集增强等各种方式，以精进大模型在特定下游任务中的表现。重点进行有针对性的数据增强，以精准提升大模型推理能力的同时，并不损失在其他任务上的表现。最后，无论在官方提供的数据集上，还是增加的测试集上，都得到了相当优秀的结果。

本次实训不仅提升了小组成员对大模型的认识，也为未来的研究和应用打下了坚实的基础。

4.2 未来工作

实验中我们仍有尚未解决的问题，其中对于一些十分简单的算数问题并无法准确回答，一些艰涩的问题却能做出很好的回答，但无法提供明确的解释。针对大模型回答问题出现的随机性和解释性缺失的问题，未来可以做更深入的研究。大模型作为一项重要的革命性的技术，要真正服务诸如亚马逊这样的大企业，可靠性仍需实践验证。

在未来学习了更多的专业知识，加强了专业相关的能力后，如果可以使用更大规模的数据集和更强的计算资源。

鸣谢　感谢朱能军老师在课程中提供的思路指导和资源支持、感谢助教的耐心教导与知识分享。

参考文献

[1] Touvron H, Martin L, Stone K, et al. Llama 2: Open foundation and fine-tuned chat models[J]. arXiv preprint arXiv:2307.09288, 2023.

[2] Hu E J, Shen Y, Wallis P, et al. Lora: Low-rank adaptation of large language models[J]. arXiv preprint arXiv:2106.09685, 2021.

[3] Wei J, Wang X, Schuurmans D, et al. Chain-of-thought prompting elicits reasoning in large language models[J]. Advances in neural information processing systems, 2022, 35: 24824-24837.

[4] Sallinen A, Boye G, Zhang M, et al. Llama-3 [8B]-MeditronV1. 0 — A State-of-the-Art Large Language Model for Medicine in 24 Hours[J].

[i] Github 仓库：https://github.com/lshAlgorithm/KDD/tree/Mygo_model