

面向大语言模型的领域知识注入与推理

说明文档

团队名称：____途普智能科技（北京）有限公司____

团队单位：____湖南大学____

成 员：____陈思 王琴 李友焕 李文杰____

2024年11月

目 录

面向大语言模型的领域知识注入与推理	1
一、赛题及赛制规则解读	3
二、基础设置	3
2.1 基座模型	3
2.2 微调框架	3
2.3 参数与显存使用说明	4
三、规则列表的生成	4
3.1 数据预处理	4
3.2 数据分析	5
3.3 检索算法实现	5
3.4 训练/推理参数设置	6
四、选项答案的生成	6
4.1 数据预处理	6
4.2 答案生成算法实现	6
4.3 训练参数设置	7

一、赛题及赛制规则解读

本赛题以规则文本的形式给定1000条应急处置规则知识和对应的实例类数据。其中规则类知识表达精准，实例类数据表达多样，要求大模型能够注入规则类数据后，具有面向下游实例类数据的推理能力。规则类知识格式为{规则ID，规则文本}。参赛者需要根据给定的多项选择题题干和候选项，选出正确的选项以及所参考的规则编号。

本赛题的评测标准由推理准确率和规则命中性能两个部分组成，两部分得分各占50%。由于两部分具有相同的权重，我们将本任务拆解为两个步骤。步骤一：通过检索策略从1000条规则中筛选出与解题最相关的10条规则；步骤二：依据选中的10条规则选出正确选项。第二章将介绍步骤一，第三章将介绍步骤二。

二、基础设置

2.1 基座模型

我们在广泛调研和考察hugging face榜单后，决定使用Qwen2.5-7B作为基座模型；使用MiniCPM-Embedding作为检索阶段的编码模型；使用MiniCPM-Reranker作为重排序模型。它们的参数总量、发布时间、训练和推理时所占显存如下表所示，均满足主办方要求的“**只能使用开源模型，基座模型选择不超过8B的开源大模型，且至少2024年10月之前公开在hugging face或其他平台。不限制模型融合，但需确保在单卡4090的24G显存服务器可以进行复现**”

表1 模型细节表

模型名称	参数量	发布时间	训练/推理显存
Qwen2.5-7B	7B	2024.9.19	<24G
MiniCPM-Embedding	2.4B	2024.2	<24G
MiniCPM-Reranker	2.4B	2024.9	<24G

2.2 微调框架

我们选择被广泛应用的LLaMA-Factory框架作为我们的微调框架，同时选择Lora作为微调方案。依据经验表明，相比于QLora和PT微调，Lora微调性能更加

稳定和高效。

2.3 参数与显存使用说明

本方案在检索阶段使用的模型总参数量为编码模型MiniCPM-Embedding（2.4B）和重排序模型MiniCPM-Reranker（2.4B）符合主办方规定的“**推理过程中，同一时刻需要挂到显存中的模型参数总量不超过8B，检索模型的参数单独计算，不占用生成答案模型的参数，但是同样不得超过8B**”。

本方案在对规则进行微调得到排序模型时，基座模型为Qwen2.5-7B，微调形成的Lora模块的参数量约为40M，训练时显存占用小于24G，推理时显存占用小于24G，符合主办方规定。

本方案在选项答案生成阶段使用的基座模型为Qwen2.5-7B，微调形成的Lora模块的参数量约为40M，训练时显存占用小于24G，推理时显存占用小于24G，符合主办方规定的“**对于生成答案的基座模型，严格限制参数不超过8B，特别是如果多个生成式模型进行后处理融合，总参数不超过8B**”。下图为各阶段参数量和显存占用量。

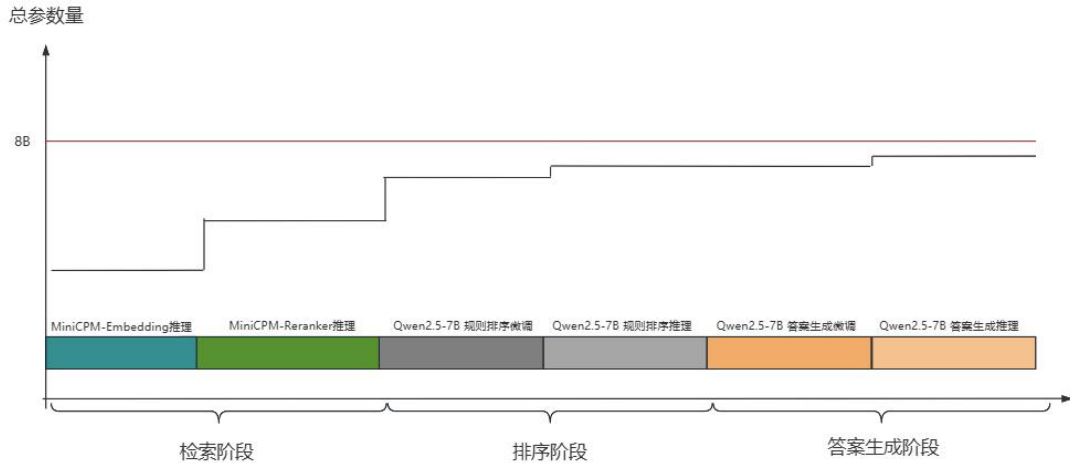


图1 各阶段总参数量

三、规则列表的生成

3.1 数据预处理

（1）规则部分：将初赛提供的`rule1.csv`与复赛提供的`复赛数据集_rule2.csv`进行合并，并转化为json格式，便于后续处理。

(2) 多选题目部分：将初赛提供的`dev`数据集中的500条数据按9: 1的比例划分为训练集（450条）和验证集（45条）。由于复赛新增训练参考集中提供了题干、候选项和参考规则，将复赛提供的复赛新增训练参考集也加入到训练集中。

3.2 数据分析

我们统计了1000条规则的平均长度为157，而Qwen2.5-7B能支持的最大输入长度为13K，因此可以支持对Top10的规则作为输入。

此外，在仔细分析所提供的规则时，我们发现规则间存在一定规律，即相似场景下的规则都聚集在一起，且尽管两个规则十分相似，如何多选题意表明了场景，那么正确规则也只包含此场景下的规则。因此，我们选择对规则进行了分类，将规则分为21类，并将类别名添加到初始规则前面，形成“主题: XXX, 内容: XXX”格式的规则数据，称之为“主题规则”。后文若不表明是主题规则，则默认表示主办方提供的初始规则。

3.3 检索算法实现

考虑到评测中涉及Hit@10指标，本阶段旨在筛选出与题目最相关的10个规则，并根据相关度由高到底排序。

(1) 粗排。我们应用MiniCPM-Embedding模型分别对题目+选项、主题规则进行编码，筛选出前20个相关主题规则。

(2) 重排序。然后利用MiniCPM-Reranker模型对这20个主题规则进行重排序，选取前10个相关主题规则。

(3) 微调。在得到10个规则后，训练出针对本赛题的排序模型。具体来说，在训练集中，输出：将正确规则放在前面，将这10个中不属于正确规则的规则依次放在正确规则后面，形成输出。输入：输入为输出规则序列随机打乱后的版本。这样微调之后，能使得模型在理解题目和其他规则的情况下，正确衡量规则对解题的重要性，这不是重排序模型能做到的。

在45条验证集上测试出最优方案后，将这45条验证集以构建训练集相同的方法加入到训练集中，形成共3500条训练数据，得到最终答案生成模型。

3.4 训练/推理参数设置

表2 检索阶段微调时的重要参数细节

参数名	参数值	参数名	参数值
Model	Qwen2.5-7B	Learning rate	1e-4
Max length	40	epoch	3
Batch size	1	bf16	16

四、选项答案的生成

4.1 数据预处理

由于在复赛阶段提供的复赛新增训练参考集中并没有提供正确选项标签，因此我们选择对初赛数据进行打伪标签操作。针对初赛测试集中的4500条数据，我们通过用不同规则列表训练出的三个Lora模块进行预测，将预测选项结果一致并且参考规则Top1相同的测试集数据作为伪数据，共计2884条。考虑伪数据并不准确，使用所有伪数据可能使得模型偏好伪数据而非真实数据，因此，我们将伪数据切分为四份，每份721条。将这四份数据与初赛划分出的训练集（450条）合并后，形成新的四份训练样本，每份样本1171条数据。

4.2 答案生成算法实现

本阶段旨在根据检索阶段得到的规则列表和题目，生成正确选项。检索阶段提供了10个规则，但在我们的试验中，将10条规则都作为上下文提供给模型将带来很大噪声，使模型性能下降，并且长度过长超出了24G显存限制。因此我们在这10条规则中选取Top5的规则加入到上下文中。此阶段的训练输入为：问题、选项和Top5条规则文本；输出为正确选项。依据四份训练样本得到四个Lora模块，

在45条验证集上测试出最优的Lora模块后，将这45条验证集以构建训练集相同的方法加入到训练样本中，共1216条训练数据，得到最终答案生成模型。

4.3 训练参数设置

表2 选项答案生成阶段微调时的重要参数细节

参数名	参数值	参数名	参数值
Model	Qwen2.5-7B	Learning rate	1e-4
Max new tokens	1	epoch	3
Batch size	1	Bf16	True