在这之前的：

1. 目前经济学研究所涉及到的大部分工作，尤其是简单的文本分类的工作其实完全不需要微调，基础的GPT模型就能很好地完成此类任务，本质的要点在于做好Prompt Engineering。

但是目前使用大语言模型的已发表文章都进行了微调工作，因此了解这一流程仍具有必要性；

1. 本人对于微调了解仅限于文本微调，且了解较浅，如有出错，请见谅

---------------------------------------------------------------------------

微调（fine tunning）是什么？

通过微调，您可以从 API 提供的模型中获得更多功能，具体功能如下：

* 比只使用提示词更高质量的结果
* 能够训练比提示中更多的示例
* 通过降低prompt用量，减少Tokens消耗
* 更快的响应速度

OpenAI 的文本生成模型已在大量文本上进行了预训练。为了有效地使用这些模型，我们在提示中包含了说明，有时还会包含几个示例。使用演示来展示如何执行任务通常被称为“少量学习”。

微调通过训练比提示中更多的示例来改进小样本学习，让您在大量任务上取得更好的结果。一旦模型经过微调，您就不需要在提示中提供那么多示例。这可以节省成本并实现更低延迟的请求。

微调的情景？

* 设定风格、基调、格式（定制大语言模型的输出方式）
* 提高产生所需输出的可靠性
* 解决原始模型无法完成复杂任务的情况
* 处理一些非常领域的特殊问题
* 执行一项难以用Prompt表达的新技能或任务

微调的步骤是什么？

1. 准备并上传训练数据
2. 训练新的微调模型
3. 评估结果并根据需要返回步骤 1
4. 使用微调模型

[百度智能云平台](https://console.bce.baidu.com/qianfan/overview)

平台优点：1.全过程可视化，使用方便；2.百度推广力度比较大，偶尔有羊毛可以薅（比如它的微调训练是全免费的）3.支持国产

平台缺点：1.虽然流程可视化，仍然新手不友好，不仔细研究完全看不懂各个部分说的什么意思；2. hand book写的莫名其妙，很容易被引导的团团转；3.注册使用后会被电话轰炸

接下来将结合百度智能云平台，就处理全流程进行说明：

**STEP1准备训练数据：**

**知识准备：**

**首先，什么是训练数据呢？**

SFT的训练数据，通常为(System,Prompt,Response)语料，其中System为选填字段。System、Prompt和Response通常指代不同的对话元素，它们的结构与作用如下：

* System: System字段的内容通常用来设定对话的背景或提供特定的指导。它定义了模型的角色、行为准则或对话的上下文。例如，系统信息可以指示模型以礼貌的方式回答问题，或者假设特定的专业领域背景。这部分有助于一致性地塑造模型的输出风格和内容。
* Prompt: Prompt的内容是用户输入的内容或问题。它是由用户提供的文本，模型需要基于此生成一个合适的响应。Prompt可以是一个问题、请求或任何其他类型的输入，模型将根据这个输入来进行推理和回答。
* Response: Response是模型生成的输出或回答。基于System设定的背景和Prompt提供的信息，模型会生成一个相应的回答。这部分是模型的实际输出，是用户与模型交互的直接结果。

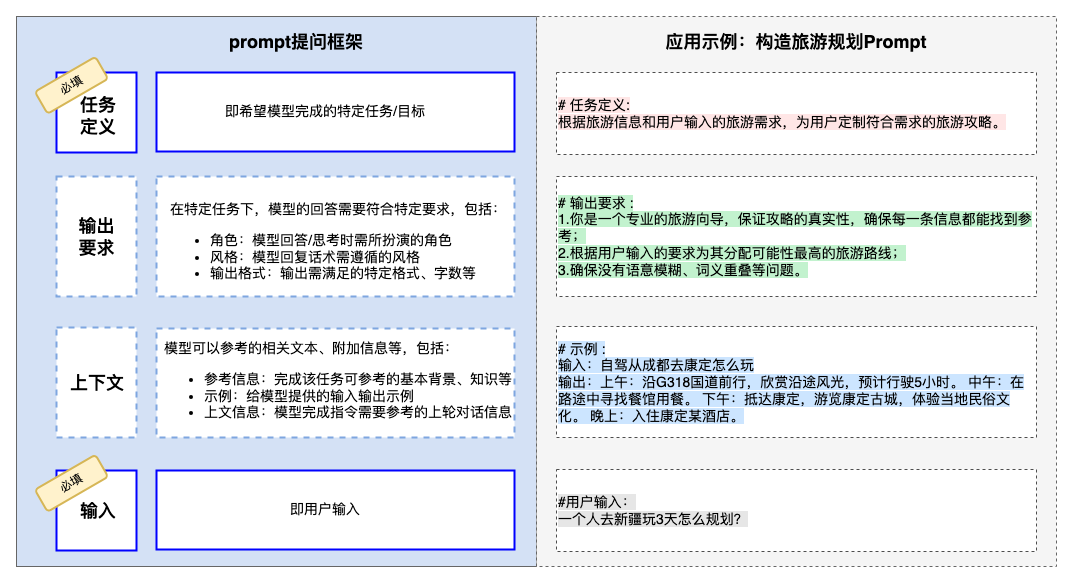
这种结构化的格式帮助在训练和实际应用中明确地传达各部分的角色和功能，从而提高交互的准确性和有效性。

**那么，什么是有价值的训练数据呢？**

* 从整体上看：
  + 在一般的场景中，通常准备**一千条**以上的训练数据进行训练，即可发挥良好的效果。
* 从单独一条的训练数据上看：
  + 请保证用户Prompt的意图清晰、数据质量优；
  + 请保证Response完全遵循用户指令，且Prompt与Response一定要一一对齐。

**那么，什么是意图清晰、数据质量优的Prompt呢？**

**一段Prompt主要由四部分组成，即任务定义、输出要求、上下文、输入。**下图将对prompt模板组成各部分进行解释，并以「为用户指定旅游攻略」这一任务为例，构造一个基础的prompt。



* 指令保证格式清晰、用词准确
  + 格式清晰，正确使用分隔符和换行、分点描述、强调词
    - 分隔符和换行：Prompt的任务、输出要求、上下文等各部分之间，请用换行符、分隔符（如###；`；【】等）进行区隔。
    - 分点描述：对模型的输出要求较多时，建议将要求进行分点描述，保证格式清晰。
    - 强调词：对于模型应当特别关注的约束条件，可以在描述前添加「注意」，表示强调；或使用markdown语法中表明加粗强调的\*\*符号进行包裹。
  + 表述清晰，可将复杂句子拆解为多个简单分句进行描述
    - 复杂句子可通过多次使用简单任务句式进行表述，以达到表述清晰的目的。
  + 表述正确，避免病句
    - 请避免Prompt中出现成分残缺、搭配不当、语序不当、句式杂糅、前后矛盾、词语误用、歧义表述 等病句，避免模型理解错误。
* 明确输出要求
  + 给模型一个适合任务的角色
    - 为模型指定一个适合目标任务的优质角色，可以让模型在回答问题时更多选择优质训练数据，以保证回答质量。指定角色时请注意应当明确角色 擅长的领域、做事原则、行为规范。
  + 明确「应该做」「不应该做」的
    - 明确约束要求：明确本次处理必须遵守或不要遵守的要求
    - 明确对参考信息的依赖程度：输出内容是严格按照材料还是根据材料自由发挥
    - 强制令遵循：通过强指令遵循，强调输出内容的语言风格，可以使用条件句、祈使句
  + 对复杂要求描述清晰详细
    - 当模型无法遵循Prompt中的某个要求时，可以尝试针对该要求添加更详细描述。
* 为模型提供参考和示例
  + 给模型一些参考信息，并引导模型使用
    - 为模型提供一些和任务相关的参考信息，并且在指令中引导模型使用这些参考信息。
  + 提供「示例」作为参考
    - 示例是提供给文心大模型的完整输入输出的样本演示。示例并不是用户必须构造的内容，但建议用户在构造Prompt时给出一到多个示例以供模型学习，具体要求如下：
    - 准确无误：示例应是有代表性的输入输出示范，且确保内容、风格、格式准确无误
    - 符合实际要求：示例中的输入输出与实际请求的输入输出应确保一致
    - 可提供多组示例：如果有多种不同的输入输出情况，可以给出多组示例
* 引导模型用思维链进行思考（CoT，Chain of Thoughts）
  + 告诉模型按特定步骤输出
    - 用户指定完成当前任务所需的步骤，这是提高模型效率和准确度的关键方法。若想向模型描述清执行步骤，需要注意：
    - 任务描述：在任务部分即说明分步骤进行思考，要求模型先根据参考信息、要求、输入等先输出阶段产出，再根据参考信息、要求、输入、阶段产出，输出最终产出。
    - 格式模板：指定完成当前任务所需的中间步骤，让模型增加思考步骤，减少输出失真。输出格式可视需求而定。
  + 引导模型在得出结论前重新思考
    - 在模型贸然得出结论之前，引导模型重新思考一遍问题，可以避免模型对问题的误判。

以下是一个prompt示例：

# 时间间隔判断任务

## 任务描述

请编写一个Prompt，用于判断两个时间点之间的小时数。需要确保格式清晰、用词准确，使模型能够准确理解并执行任务。

## 输出要求

1. \*\*角色设定\*\*：

- 你是一位专门处理时间计算的智能助手，擅长时间管理和日历计算。

- 你的首要目标是准确计算时间间隔，以小时为单位。

2. \*\*具体要求\*\*：

- \*\*应该做\*\*：

- 计算两个时间点之间的总小时数。

- 提供详细的计算过程。

- 输出结果需以用户友好的方式呈现。

- \*\*不应该做\*\*：

- 不要忽略任何时间单位的转换。

- 不要提供与时间间隔无关的信息。

3. \*\*参考信息依赖\*\*：

- 严格根据用户输入的两个时间点进行计算。

## 示例

\*\*示例输入\*\*：

- 开始时间：2023年10月1日 08:00:00

- 结束时间：2023年10月3日 10:30:45

\*\*示例输出\*\*：

### 小时数计算

- 总小时数：50小时

### 计算过程

1. 从2023年10月1日 08:00到2023年10月2日 08:00，共计24小时。

2. 从2023年10月2日 08:00到2023年10月3日 08:00，再加24小时。

3. 从2023年10月3日 08:00到2023年10月3日 10:30，共计2小时。

4. 综合计算：24小时 + 24小时 + 2小时 = 50小时。

## 思维链引导

请按照以下步骤进行思考和计算：

1. \*\*第一步\*\*：根据输入的开始和结束时间，识别时间间隔。

2. \*\*第二步\*\*：将时间间隔转换为小时数，确保计算准确。

3. \*\*第三步\*\*：综合计算总小时数。

4. \*\*第四步\*\*：检查计算过程，确保无误后输出结果。

通过以上步骤，确保准确得出总小时数，并以清晰的格式呈现给用户。

### 输入

**那么，什么是质量高的Response呢？**

* 确保数据中Response严格遵循指令、示例要求：必须保证指令中的每一条限定要求，在回复中均被满足（即便指令中使用了「你可以」「建议你」此类模糊的修饰）。
* 确保回复的事实准确性：确保回复中的每个客观事实均正确、无幻觉。注意，数据不仅要符合业务逻辑，也要符合普遍常识。
* 描述简明易懂：复杂任务描述语句可拆解为多个简单短句，便于描述和理解。
* Response内容表述正确：无重复和冗余、语言表达正确、逻辑自洽、标点使用正确。
* 确保回复的价值观正确、安全合规。

上面的内容为Response的基础原则，距离质量高的Response还有一定差距，质量高的Response应具备：

* 提升内容丰富度：提升回复内容的全面性、丰富性、格式优美性、语言吸引力等。
* 合理分布数据比例，提升数据的覆盖面（尤其是边界情况）：确保数据的分布合理，比如正负例比例，不同的任务分支出现的比例。类比于代码测试，数据也应该尽可能覆盖到每一个分支和每一种边界情况。

**实操：**

1.数据标注与准备：有能力的情况下，应该以严格的人工标注或者现成的自然数据为准；但是对于普通的学生，人工大量标注数据显然不现实，可以先用能力更强的的AI生成预训练数据集。并整理成规范格式

2.数据集创建：进入智能云平台<https://console.bce.baidu.com/qianfan/overview>

选择“数据集 > 通用数据集 > 创建数据集”

目前平台提供了两种可用的训练数据格式，个人比较推荐后者，其与传统的API调用所设置的user，assistant逻辑相同，可以在后续实际调用中继续利用。按照平台提示，上传文档即可图形用户界面, 文本, 应用程序, 聊天或短信

描述已自动生成

3.数据清洗：一般建议在上传前就本地写代码清洗一下你的数据，一般建议删除符号，数字，链接等无效内容。不过百度云平台也提供了一个数据清洗的功能，但个人实测并不好用

**STEP2训练新的微调模型**

**知识准备：**

**SFT还是Post-pretrain？**

Post-pretraining的目标是在不过度专化到某个特定任务的同时，提高模型对特定领域的理解和表现。

SFT的目的是使模型更好地适应特定的任务或领域【垂直领域】，比如特定类型的语言理解或生成任务。

对于实操领域的分类任务，应该使用SFT。

**全量更新还是LoRA？**

全量更新在训练过程中对大模型的全部参数进行更新

LoRA在固定预训练大模型本身的参数的基础上，在保留自注意力模块中原始权重矩阵的基础上，对权重矩阵进行低秩分解，训练过程中只更新低秩部分的参数。

实操中差别不大，LoRA消耗Tokens更少。

**迭代轮次**

迭代轮次（Epoch），控制模型训练过程中遍历整个数据集的次数。建议设置在1-5之间，小数据集可增大Epoch以促进模型收敛。

对于1-2千条，可以设置为2-3来增强学习效果，如果达到2-3万则只要是1就行

**选择8K还是更大？**

取决于单条训练数据是否超过8192 tokens

**实操：**

进入智能云平台<https://console.bce.baidu.com/qianfan/overview>

选择“模型精调 > SFT > 创建训练作业”

如果只是测试数据集效果，可以选择Speed，数据集选择已经上传到云端的数据。其他数值皆可选择默认，后续根据训练结果进行修改，不过迭代轮次（Epoch）可根据实际情况提前决定。

**选择闲时调度可以免费微调**

**STEP3评估结果**

**知识准备：**

**Loss指标**

训练过程中，可以查看Training Loss和Validation Loss可用于评估训练任务的效果。

* Training Loss：对比模型在训练集上的预测值和真实值的差异，考察模型在训练集上的拟合情况。
* Validation Loss：对比模型在验证集上的预测值和真实值的差异。

对于Loss曲线，我们需要关注它的平滑度、收敛度、泛化性。

* 平滑度：平滑的曲线意味着随着训练的进度的增加，模型的效果是逐步变化并且有一致性的。指标平滑下降说明模型的效果是稳定提升的。如果曲线中存在陡增或者陡降，说明存在异常情况。
* 收敛度：收敛说明曲线达到稳定或者最优的状态。我们期望的状态是曲线收敛到一个点，当训练增加时指标不会增加。
* 泛化性：好的曲线需要具有泛化能力，即不仅在训练集上有好的效果，在验证集上表现同样优异。如果在训练集和验证集上差异较大，说明训练不具备泛化性。如果在训练集上表现良好，但是在验证集的新数据上表现差，说明模型存在过拟合。

通常来说，随着训练过程的增加，期望的曲线是Training Loss和Validation Loss同时稳定平滑下降，并且收敛到一个相对低的值。下面我们分别分析几种常见的Loss曲线。

**Loss曲线分析**

1、收敛-Loss曲线同时平缓

Training Loss和Validation Loss同时下降，并且到最后值趋于稳定，是比较理想的状态。Training Loss下降表明模型有效的学习到训练数据，Validation Loss下降表示模型的泛化能力较强。

图表, 折线图

描述已自动生成

**其他指标：**

BLEU-4：用于评估模型生成结果与标注结果之间的匹配程度，计算 1-gram 到 4-gram 的加权平均精确率。ROUGE-1：评估模型生成结果与标注结果的 1-gram 召回率。ROUGE-L：基于最长公共子序列（LCS）计算模型生成结果与标注结果的召回率。

**实操：**

1.基础训练数据：

选择“模型精调 > SFT > 找到对应的作业 > 详情 > 评估报告”

默认评估报告给如下指标：BLEU-4、ROUGE-1、ROUGE-2、ROUGE-L、Embedding 距离、编辑距离

由于我们选择的是文本分类任务，对于结果的精准度要求非常高，因此合理的值应该接近于100

2. Perplexity（困惑度）Loss曲线图

一般而言，应该在训练2-3轮次后下降到稳定

3.实操比较

由于指标不能真正反映实际情况，无法排出过拟合的情况。我们还可以通过发布模型进行实操比较

发布模型：选择“在线推理 > 我的服务 > 创建服务”

模型比较：在模型发布成功后，选择“体验中心”

参考教程：

Prompt工程<https://platform.openai.com/docs/guides/prompt-engineering>

微调：[https://platform.openai.com/docs/guides/fine-tuning](https://platform.openai.com/docs/guides/fine-tuning#use-a-checkpointed-model)

数据集：<https://cloud.baidu.com/doc/WENXINWORKSHOP/s/5m36rbvze>

模型微调：<https://cloud.baidu.com/doc/WENXINWORKSHOP/s/Oliu6n43o>

评估结果：<https://cloud.baidu.com/doc/WENXINWORKSHOP/s/Sm36rj43i>