指令微调数据集

数据集怎么准备?

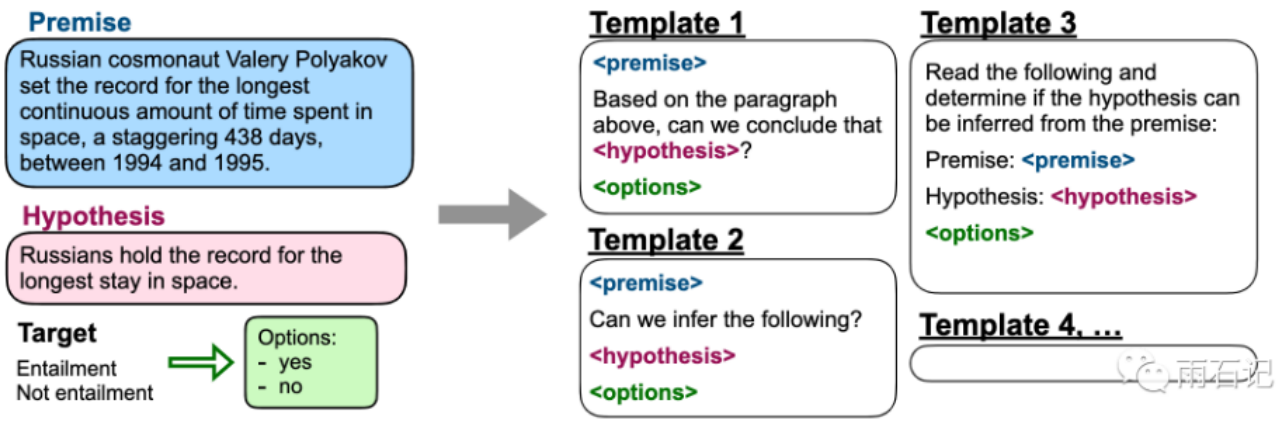
1、使用huggingface上已有的用于微调的数据集

2、使用开源LLM使用的数据集

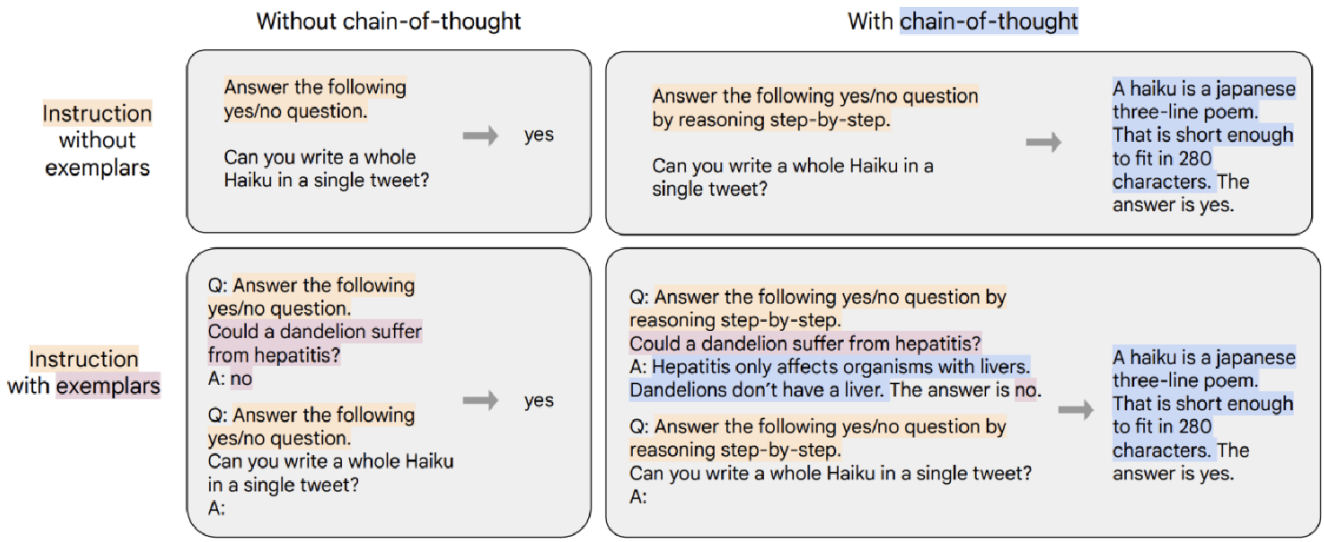
3、在现有的数据集上加上一些模板

举例:

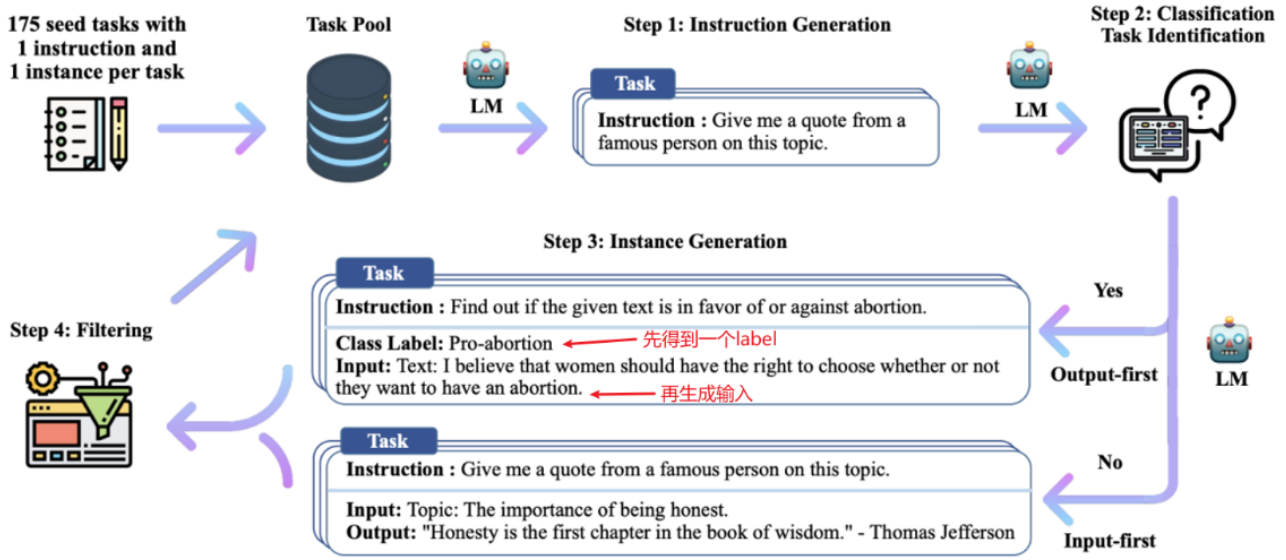
对于一个情感分类数据集，将它变成生成电影影评的任务。为了增加多样性，这里使用了多种模板。



加上**COT**:



self-Instruct（用GPT来构造）：



总共四步：1）指令生成，2）识别指令是否代表分类任务，3）用输入优先或输出优先的方法生成实例，4）过滤低质量数据。

这个过程可以迭代地重复许多次，直到达到大量的任务。

在step2中：做了一个是否是分类任务的判断，如果是，就依据得到的label生成相应的input，这么做可以提高得到数据的多样性。

在step4中为了保证多样性的一些做法：

只有当一条新指令与任何现有指令的ROUGE-L重叠度小于0.7时，才会被添加到任务池中。

排除了包含一些特定关键词（如图像、图片、图表）的指令，这些关键词通常不能被语言模型处理。

在为每条指令生成新的实例时，过滤掉那些完全相同的实例或那些具有相同输入但不同输出的实例。

> ROUGE-L衡量参考文本和生成文本之间最长公共子序列的相似性。
>
> 精确度：$P = \frac{LCS(X,Y)}{|X|}$ 召回率：$R = \frac{LCS(X,Y)}{|Y|}$ F1 score：$F1 = \frac{2 \cdot P \cdot R}{P+R}$ 就是ROUGE-L的值 其中，$LCS(X,Y)$是序列 X 和 Y 之间的最长公共子序列的长度，|X|和 |Y|分别是序列X和 Y的长度。

补充：llama factory中使用的一些模型评估指标

指标	含义
BLEU-4	BLEU（Bilingual Evaluation Understudy）是一种常用的用于评估机器翻译质量的指标。BLEU-4 表示四元语法 BLEU 分数，它衡量模型生成文本与参考文本之间的 n-gram 匹配程度，其中 n=4。值越高表示生成的文本与参考文本越相似，最大值为 100。
predict_rouge-1 和 predict_rouge-2	ROUGE（Recall-Oriented Understudy for Gisting Evaluation）是一种用于评估自动摘要和文本生成模型性能的指标。ROUGE-1 表示一元 ROUGE 分数，ROUGE-2 表示二元 ROUGE 分数，分别衡量模型生成文本与参考文本之间的单个词和双词序列的匹配程度。值越高表示生成的文本与参考文本越相似，最大值为 100。
predict_rouge-l	ROUGE-L 衡量模型生成文本与参考文本之间最长公共子序列（Longest Common Subsequence）的匹配程度。值越高表示生成的文本与参考文本越相似，最大值为 100。
predict_runtime	预测运行时间，表示模型生成一批样本所花费的总时间。单位通常为秒。
predict_samples_per_second	每秒生成的样本数量，表示模型每秒钟能够生成的样本数量。通常用于评估模型的推理速度。
predict_steps_per_second	每秒执行的步骤数量，表示模型每秒钟能够执行的步骤数量。对于生成模型，一般指的是每秒钟执行生成操作的次数。

有一段时间，人们认为数据集的质量 >> 数量，但是llama3的出现打破了这一印象。因此数据集的质量和数量同等重要。

如何保证数据集的质量?

可以通过以下指标来过滤生成的数据集以保证质量：

Instruction Length：指令的长度。

困惑度(perplexity)：它衡量了模型对自己预估结果的不确定性，低困惑度说明模型对自己很自信，但是不一定准确，但是又和最后任务的表现紧密相关。因此通过预训练模型计算回复的困惑度作为复杂性指标，困惑值越大意味着数据样本越难。

> 句子生成的概率：
>
> $P(w_1, w_2, \ldots, w_N) = \prod_{i=1}^{N} P(w_i | w_1, \ldots, w_{i-1})$
>
> 其中，$P(w_i | w_1, \ldots, w_{i-1})$是模型给定前i-1个词时第i个词的条件概率。
>
> 使用对数概率：
>
> 困惑度$PPL = e^{\frac{1}{N} \sum_{i=1}^{N} \log P(w_i | w_1, \ldots, w_{i-1})}$
>
> 这里的 N 是句子中的词数。

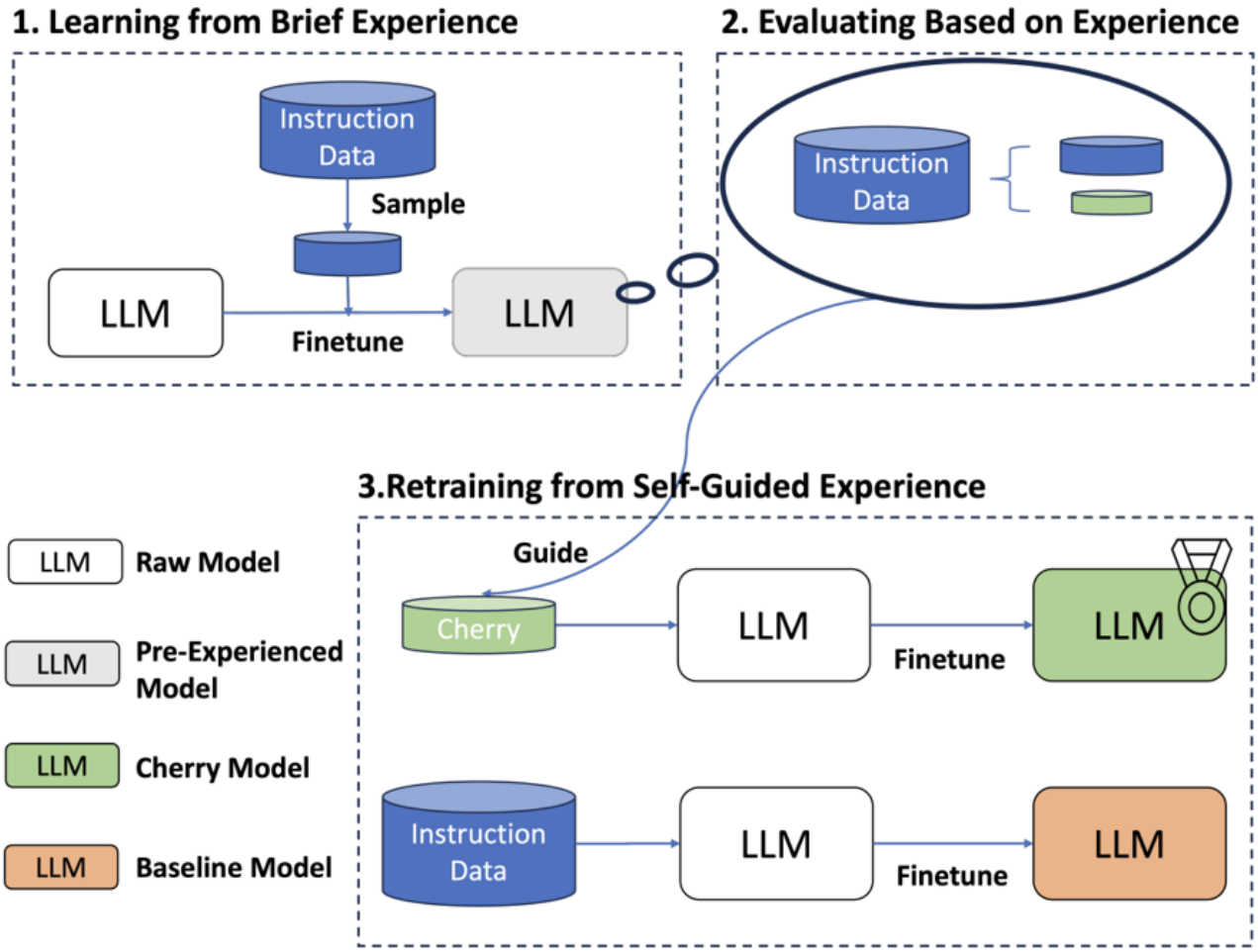
论文：When Less is More: Investigating Data Pruning for Pretraining LLMs at Scale 提出的另外两个指标错误L2范数(ErrorL2-Norm)、记忆化(memorization)效果不如困惑度。

Direct Scoring：直接让ChaGPT给指令的复杂性打分。

Instruction Node：利用ChatGPT将指令转换成语义树，通过树的节点数作为复杂性指标。

Instag Complexity：利用ChatGPT对部分数据进行打标签，再训练一个Llama模型，再利用训练后的Llama模型对全量数据预测，标签越多说明数据越复杂。

IFD：指令跟随难度作为复杂性指标。



1、利用少量进行进行模型初学

2、利用初学模型计算原始数据中所有IFD指标;

3、利用筛选出的优质数据（樱桃数据）进行模型重训练。

条件回答分数（ Conditioned Answer Score，CAS）：利用初学模型可以对数据集中所有样本进行预测，通过指令内容预测答案内容，并可以获取预测答案与真实答案直接的差异值（利用交叉熵），即CAS。根据CAS的高低，可以判断出模型对指令Q生成答案A的难易，但也可能受到模型生成答案A的难易程度的影响。

直接答案分数（Direct Answer Score，DAS）：利用模型直接对答案进行续写，再根据答案真实内容获取直接的差异值，即DAS。DAS得分越高，表明该答案对模型生成来说本身就更具挑战性或更复杂。

CAS

$$L_\theta(A|Q) = -\frac{1}{N}\sum_{i=1}^{N}\log P(w_i^A|Q, w_1^A, w_2^A, \ldots, w_{i-1}^A; \theta)$$

$$\mathrm{IFD}_\theta(Q, A) = \frac{s_\theta(A|Q)}{s_\theta(A)}$$

DAS

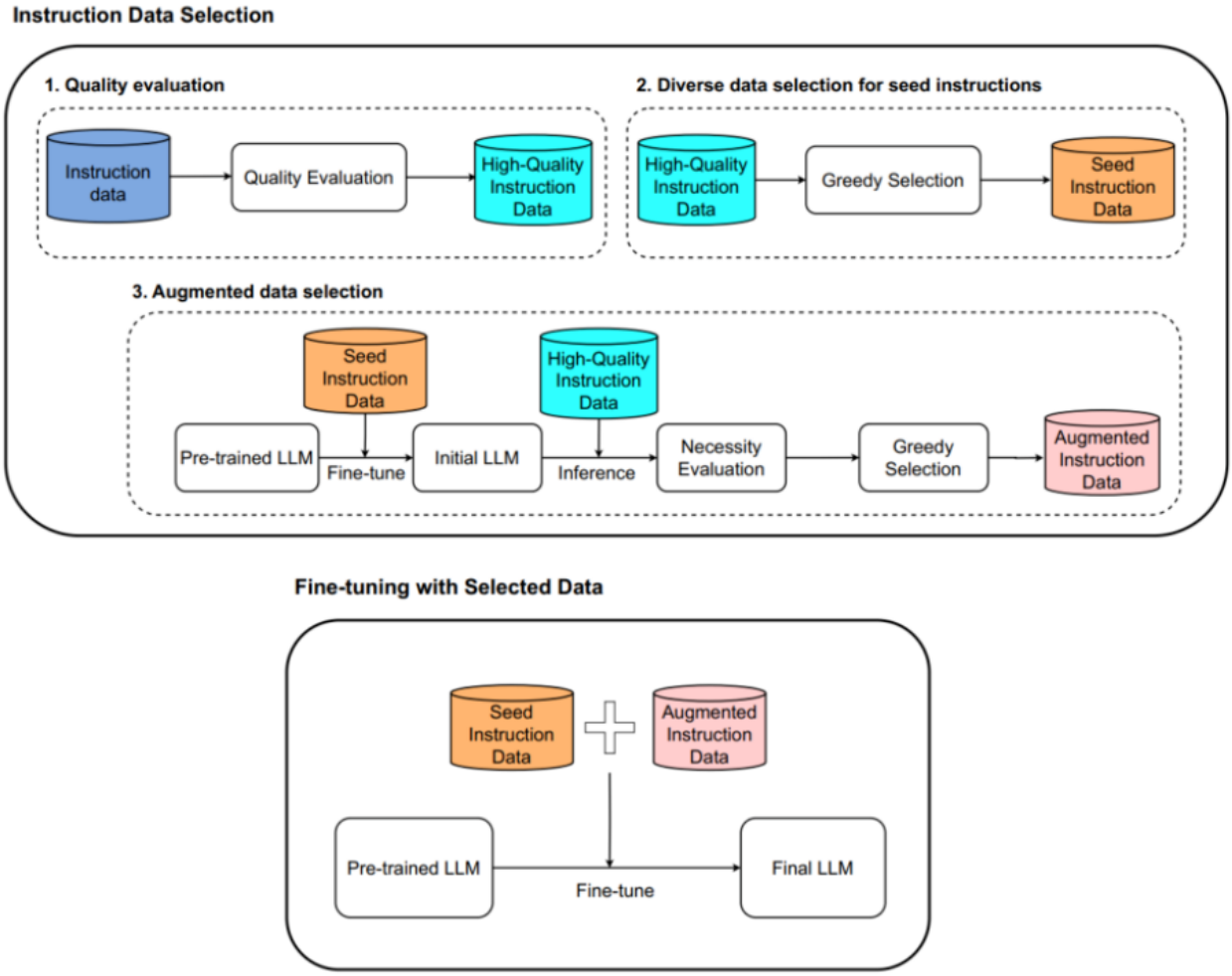
$$s_\theta(A) = -\frac{1}{N}\sum_{i=1}^{N}\log P(w_i^A|w_1^A, \ldots, w_{i-1}^A; \theta).$$

利用IFD指标对数据进行筛选，减缓了大模型对答案本身拟合能力的影响，可以直接衡量给定指令对模型生成答案的影响。

较高的IFD分数表明模型无法将答案与给定的指令内容进行对齐，表明指令的难度更高，对模型调优更有利。

MoDS

MoDS方法主要通过质量、覆盖范围、必要性三个指标来进行数据的筛选。如下图所示：



质量筛选：

对于数据进行质量过滤时，采用OpenAssistant的reward-model-debertav3-large-v2模型（一个基于DeBERTa架构设计的奖励模型）对数据进行质量打分。将原始数据的Instruction、Input、Output的三个部分进行拼接，送入到奖励模型中，得到一个评分，当评分超过α时，则认为数据质量达标，构建一份高质量数据集-Data1。

多样性筛选：

为了避免所选质量数据高度相似，通过K-Center-Greedy算法进行数据筛选，在最大化多样性的情况下，使指令数据集最小。获取种子指令数据集（Seed Instruction Data）-SID。其中，用BERT模型为指令数据生成句向量来计算不同数据之间的距离。

Algorithm 1 K-Center-Greedy (Sener and Savarese, 2017)

Input: data x_i, existing pool s^0 and a budget b

Initialize $s = s^0$

repeat

$u = argmax_{i \in [n] \setminus s} min_{j \in s} \Delta(x_i, x_j)$

$s = s \cup u$

Until $|s| = b + |s^0|$

return $s \setminus s^0$

必要性筛选：

不同的大型语言模型在预训练过程中所学到的知识和具有的能力不同，因此在对不同的大型语言模型进行指令微调时，所需的指令数据也需要不同。

对于一条指令，如果给定的大型语言模型本身能够生成较好的回答，则说明给定的大型语言模型具有处理该指令或者这类指令的能力，反之亦然，并且**那些不能处理的指令对于模型微调来说更为重要。**

- 使用SID数据集对模型进行一个初始训练
- 用训练好的初始模型对整个高质数据集-Data1中的指令进行结果预测
- 利用奖励模型对结果进行评分，当分值小于β时，说明初始模型不能对这些指令生成优质的回复，不具有处理这些类型指令的能力，获取必要性数据集-Data2
- 对Data2进行多样性筛选，获取增强指令数据集（Augmented Instruction Data）-AID