【手撕RLHF-Rejection Sampling】如何优雅的从SFT过渡到PPO

小冬瓜AIGC
原创课程➡️公众号：手撕LLM

来自专栏 · 手撕LLM >

> 我是小冬瓜AIGC，原创超长文知识分享，已帮助多名同学速成上岸LLM赛道
> 研究方向：LLM、RLHF、Safety✦、Alignment✦

Pre-Requirements

本文需要具备系统的LLM知识，掌握RLHF-PPO为前提阅读会非常丝滑。

手撕RLHF专题

如果你对以下topic感兴趣，可以持续关注

1. Rejection Sampling 算法原理，使用RS有什么优势
2. 哪些方案涉及到多轮RLHF，如何做多轮RLHF
3. 如何做RLHF数据工程，如何进行偏好标注，数据如何配比
4. 为什么要多轮iterative reward model？为什么需要两个Reward Model✦
5. LLM Safety如何从什么维度评估？如何评估LLM的幻觉、偏见和毒性
6. PPO中 Safety/helpful reward model如何协同
7. Ghost Attention✦是什么？多轮数据数据格式是怎么样的？如何提升多轮对话能力？代码如何实现
8. Context Distillation 如何工作，为何能增加LLM安全性
9. 如何组织红队攻击？Adversial Attack Prompt是什么，有哪些攻击类型？
10. chatGPT、Antropic和LLaMA 2✦的 RLHF方案有什么区别
11. 什么是refusal reply？如何减少“误拒答”？如何做问答模版
12. 还有哪些RLHF方案？如何不用reward model做RLHF，如何在Pretraining阶段做RLHF
13. PPO调试和训练有哪些tricks

---

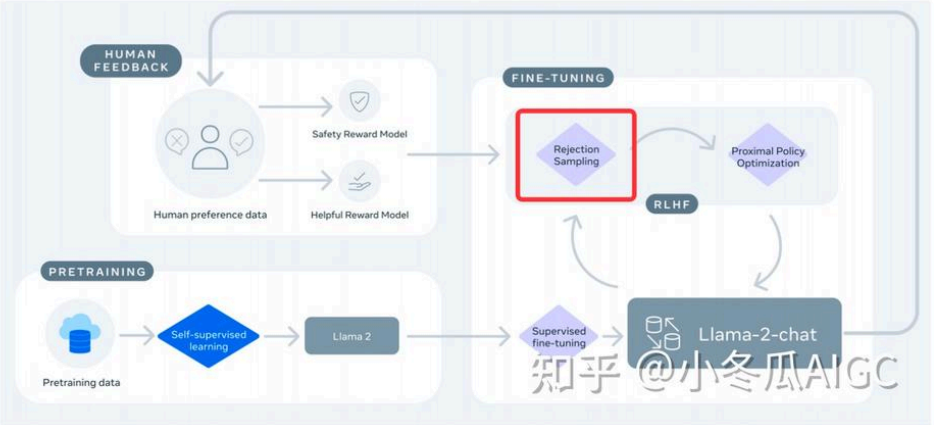
1. Rejection Sampling Fine-Tuning总览

> Llama 2: Open Foundation and Fine-Tuned Chat Models
> Meta✦

在 `LLaMA2` 里存在 `Rejection Sampling`，非常让人困惑，我早期以为是采样定理里面的拒采样，直至后来理解了相关的 `RLHF` 方案后才搞清楚 `RS`

1. 在 `LLaMA2` 中主要的 `RLHF` 微调方法一个是 `PPO` (RLHF V5), 一个是 `Rejection Sampling` (RLHF V1-4)
2. `Rejection Sampling` 并非采样定理里面的拒采样，而是一种 `Best-of-N` 采样N个生成结果, 计算对应的 `reward`
3. 通过 `reward score` 取出分数最大的生成结果，进行 `SFT`，此时比较简单的(相较 `PPO`)做对齐(alignment)
4. 在 `RLHF V4` `PPO` 之前一直用 `Rejection Sampling`

> Until RLHF (V4), we used only Rejection Sampling fine-tuning, and after that, we combined the two sequentially, applying PPO on top of the resulted Rejection Sampling checkpoint before sampling again

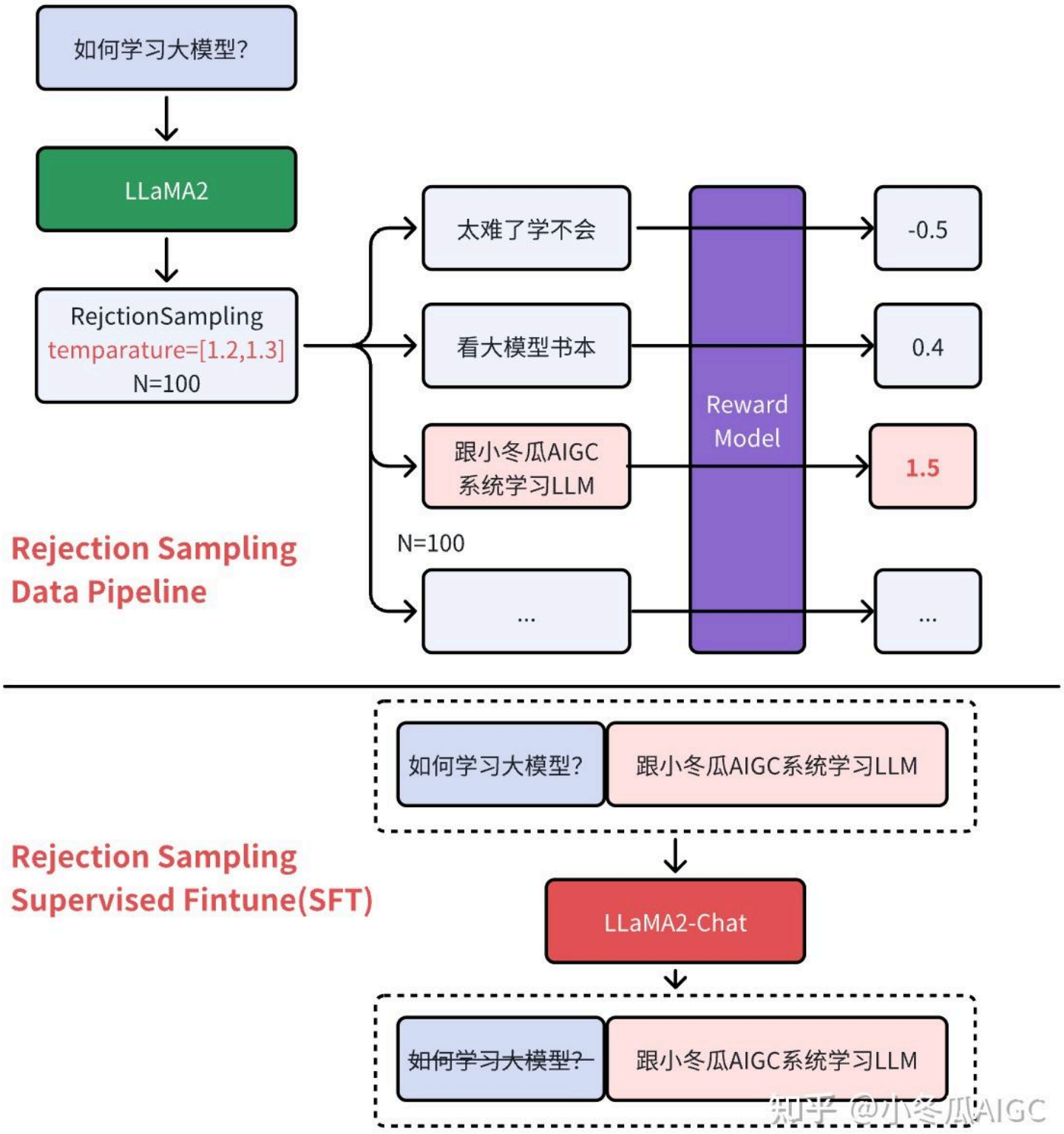


2. Rejection Sampling 数据和微调流程图解

需要了解Genreate 采样可回顾小冬瓜AIGC：【手撕LLM-Generation】Top-K+重复性惩罚

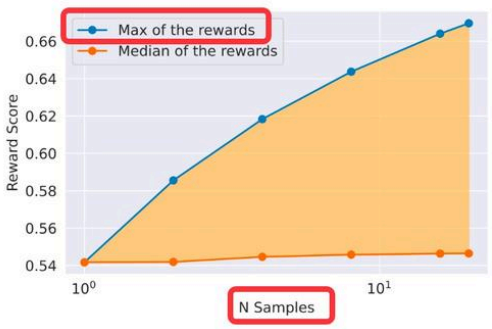
1. 对模型进行N个生成采样(`Generate`)，取最大 `reward` 的 `response` 作为 `Rejection Sampling fine-tuning` 的数据
2. 随着 `sampling` 数量 `N` 越多，`max reward` 会越高
3. 采样时可以随机设置不同的温度 $T \in [1.2, 1.3]$ ，其生成的结果多样性(diversity)越好

> the optimal temperature when sampling between 10 and 100 outputs is $T \in [1.2, 1.3]$.



3. Rejection Sampling结果分析

具体分析不同的generate参数N,T 和对应的reward分值



随着采样数量增加到100, Max Reward提升0.54-> 0.7

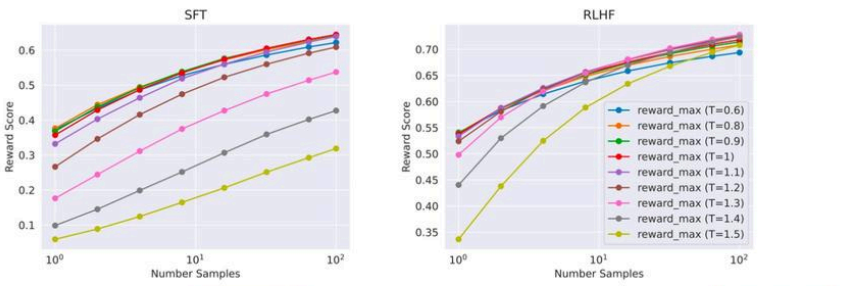


Figure 8: RLHF impact of the temperature when sampling N outputs and scoring them with a reward model

不同的T值会影响模型输出的准确性，采样数量小时和T=1.5时，reward score分数非常低

4. Rejection Sampling相关策略

- 通常大模型采样得到的 `Rejction Sampling data` 同样可以用来蒸馏(`distill`)训练小模型
- `Rejection Sampling` 的评价依赖 `reward model` 这样可以通过RS与RM进行迭代提升
- 采样策略有两种，一种是采样最近的模型，一种采样之前所有的模型 如RLHFV1/V2

> we modified our strategy, incorporating top-performing samples from all prior iterations, such as those used in RLHF-V1 and RLHF-V2.

5. Rejection Sampling Generate实现

使用Tranformers库实现Rejection Sampling，其背后可以通过采样generate采样实现。

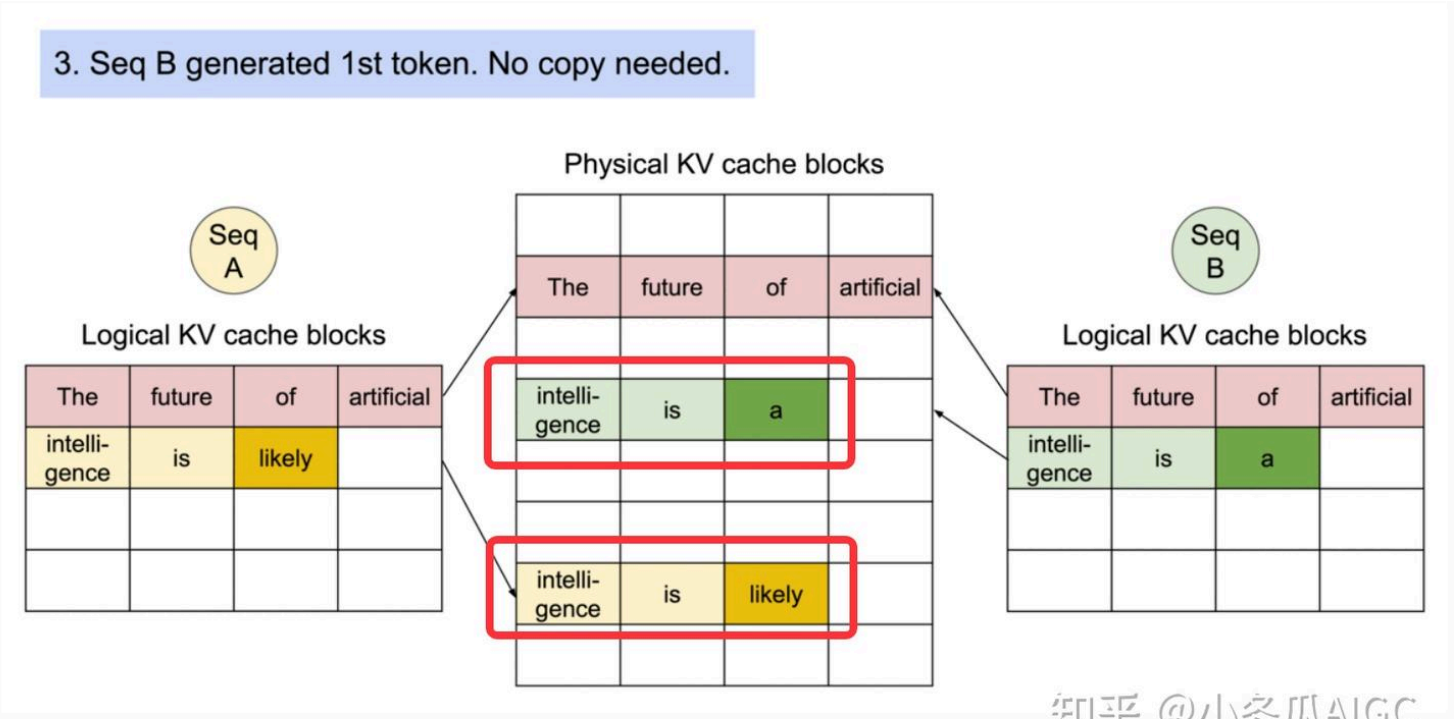
```
from transformers import AutoModelForCausalLM, AutoTokenizer
set_seed(12)
rejection_sample_n = 3
tokenizer = AutoTokenizer.from_pretrained("llama") # your tokenizer
model = AutoModelForCausalLM.from_pretrained("llama") # your model
model_inputs = tokenizer('I enjoy walking with my cute dog', return_tensors='pt')
sample_outputs = model.generate(
    **model_inputs,
    max_new_tokens=40,
    do_sample=True,
    top_k=50,
    top_p=0.95,
    temparture=1.2, # random 1.2~1.3
    num_return_sequences=rejection_sample_n,
)
for i, sample_output in enumerate(sample_outputs):
  print("{}: {}".format(i, tokenizer.decode(sample_output, skip_special_tokens=True)))
```

结果

```
0: I enjoy walking with my cute dog, but there's another side to that. Because I am a single mom, and the fact that I am doing this ki
nd of thing for my dog has caused me a lot of stress. Especially since
1: I enjoy walking with my cute dog and always try not to let the dog get close to me. I would also suggest getting to know him better
as he loves to be led by me and his paws which help in moving me. If
2: I enjoy walking with my cute dog, and she's never been a shy, sweet, or carefree person in her life. She loves to be hugged, and
I'm so sorry we never found that out."
On Facebook
```

为提升采样效率，可以借助 `PageAttention` 进行并行采样

无法贴gif, 具体看 vLLM: Easy, Fast, and Cheap LLM Serving with PagedAttention



Example generation process for a request that samples multiple outputs.

6. Rejection Sampling Reward实验对比

1. 对比a. `ref_model`，b. `model(RLHF)` c. `ref_model+RejectionSampling`
2. 期望 `ref+RS` 的 `reward` 会比 `ref` 要好，同时 `ref+RS` 结果接近 `RLHF`

加载模型

```
model = AutoModelForCausalLMWithValueHead.from_pretrained(model_name)
ref_model = AutoModelForCausalLMWithValueHead.from_pretrained(ref_model_name)
reward_pipe = pipeline("sentiment-analysis", model=reward_model, device=device)
```

采样

```
# Rejection Sampling参数
gen_kwargs = {"min_length": -1, "top_k": 0.0, "top_p": 1.0, "do_sample": True, "pad_token_id": tokenizer.eos_token_id}

for i in range(bs):
    gen_len = output_length_sampler()
    query = torch.tensor(query_tensors[i])

    output = ref_model.generate(query.unsqueeze(dim=0).to(device), max_new_tokens=gen_len, **gen_kwargs).squeeze()
    response_tensors_ref.append(tokenizer.decode(output)) # result1: ref_model

    output = model.generate(query.unsqueeze(dim=0).to(device), max_new_tokens=gen_len, **gen_kwargs).squeeze()
    response_tensors.append(tokenizer.decode(output)) # result2: model(RLHF)

    queries = query.repeat((Rejection_Sample_N, 1)) # ref_model+RejectionSampling
    output = ref_model.generate(queries.to(device), max_new_tokens=gen_len, **gen_kwargs).squeeze()
    response_tensors_best_of.append(tokenizer.batch_decode(output))
```

Reward 评分

```
# reward_pipe is seq classification
scores_ref = [output[0]["score"] for output in reward_pipe(response_tensors_ref, **sent_kwargs)]
scores = [output[0]["score"] for output in reward_pipe(response_tensors, **sent_kwargs)]
scores_best_of = []
```

```
for i, response in enumerate(response_tensors_best_of):
    # base_score = scores_ref[i]
    scores_best_of.append(torch.tensor([output[0]["score"] for
                          output in reward_pipe(response, **sent_kwargs)]))
output_data["scores (best_of)"] = [a.max().item() for a in scores_best_of]
```

结果符合预期

`ref+RS` 的 `reward` 会比 `ref` 要好，同时 `ref+RS` 结果等于 `RLHF`

	query	response (ref)	scores (ref)	response (RLHF)	scores (RLHF)	response (best_of)	scores (best_of)
0	I'm a pretty old	I'm a pretty old kid, well, with lots of girl	1.179652	I'm a pretty old lady, and I loved this movie ...	2.218363	I'm a pretty old, stinking,acting kinda chick ...	2.016955
1	One of the most	One of the most psychologically devastating as...	2.477277	One of the most Antibiotic Apps I have seen in	2.145479	One of the most memorable performances of this...	2.676944
2	Okay, as	Okay, as ruthless as they are, even their leve...	1.466462	Okay, as I enjoyed the movie. It's added bonus...	2.239827	Okay, as I put it in such a negative mood, it ...	1.478424
3	Watching "Kro	Watching "Kroger" (1915-	0.186047	Watching "Kroven". The film has a	1.044690	Watching "Kro" is an entertainment craze	1.389495
4	Seriously what were they thinking?	Seriously what were they thinking? It ain't go...	1.010697	Seriously what were they thinking? It's a very...	2.753088	Seriously what were they thinking? It was stil...	2.523514
5	OK Hollywood	OK Hollywood goes into a total game of audio, ...	0.934041	OK Hollywood shoot, and this is a classic. Som...	2.517364	OK Hollywood pay and the freaky set-up of this...	1.634765
6	"Bend It	"Bend It, Luther, Dodge, Church Goes to Rome w...	0.039218	"Bend It all" is a sophisticated, drawing and ...	2.583935	"Bend It 9"/"Zara Pephoto") and an honest, rea...	2.557210
7	While the premise behind The House	While the premise behind The House of Dracula ...	-0.079306	While the premise behind The House Intelligenc...	0.205217	While the premise behind The House of Dracula ...	1.576589

7. Anthropic：Rejection Sampling

> Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback
> Anthropic

更早期Anthropic的RLHF工作里也采用Rejection Sampling， 可以认为RS就是一种数据增广方式

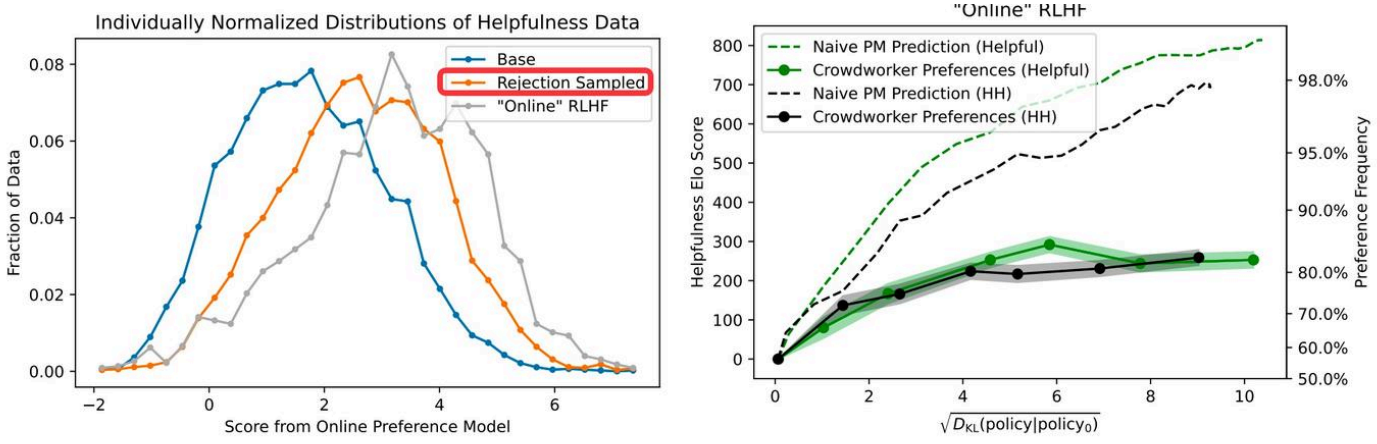


Figure 15 **(left)** This plot shows individually normalized distributions of held-out helpfulness data from our base dataset (mostly with context-distilled models), from models augmented with rejection sampling, and from data collected with our iterated 'online' RLHF models. The upper tail of the distribution receives far more support from the RS and online models, which should make it possible for preference models to learn more subtle distinctions among high-quality responses, and amplify the value of further data collection. **(right)** We compare helpfulness Elo scores of our HH and pure-helpfulness iterated online RLHF models at various points during RLHF training. Note that Elo scores and preference frequency are measured relative to the initial snapshot, which is our 52B context distilled model in both cases. Elo scores in both subplots only evaluate helpfulness.

8. 总结

1. `Rejection Sampling` 本质上是一种LLM采样进行增广数据的方法，通过 `Reward Model` 可以获得高 `reward` 的 `SFT` 数据
2. RLHF里的标注成本目前非常模糊且存在一致性问题，所以越来越多的 `RLHF` 数据 `colletion` 朝向无标注的数据工程
3. `RS` 是介于 `SFT` 和 `PPO` (或其他 `RL` 方法)之间的过渡方法，其目的是为了减轻 `ppo` 的训练难度,使对齐更加平滑，同时可以迭代提升 `reward model`
4. 代码实现其实很简单，只要会 `Generate Sampling` 就可以做 `RLHF`
5. 数据增广拓展：在 `LLM-QAT` 里，也通过采样得到蒸馏训练所需要的样本

《手撕RLHF》 解析如何系统的来做LLM对齐工程

小冬瓜AIGC：【手撕RLHF-Safe RLHF】带着脚镣跳舞的PPO

小冬瓜AIGC：【手撕RLHF-Rejection Sampling】如何优雅的从SFT过渡到PPO

《手撕LLM》 系列文章+原创课程：LLM原理涵盖Pretrained/PEFT/RLHF/高性能计算

小冬瓜AIGC：【手撕LLM-QLoRA】NF4与双量化-源码解析

小冬瓜AIGC：【手撕LLM-RWKV】重塑RNN 效率完爆Transformer

小冬瓜AIGC：【手撕LLM-FlashAttention】从softmax说起，保姆级超长文！！

小冬瓜AIGC: 【手撕LLM-Generation】Top-K+重复性惩罚

小冬瓜AIGC：【手撕LLM-KVCache】显存刺客的前世今生--文末含代码

小冬瓜AIGC：【手撕LLM-FlashAttention2】只因For循环优化的太美

《手撕Agent》 从代码和工程角度，探索能够通向AGI的Agent方法

小冬瓜AIGC：【手撕Agent-ReAct】想清楚再行动、减轻LLM幻觉

> 我是小冬瓜AIGC，原创超长文知识分享，原创课程已帮助多名同学速成上岸LLM赛道
> 研究方向：LLM、RLHF、Safety、Alignment

编辑于 2023-12-06 18:08 · 广东

LLM 大模型 RLHF