

5810 Final Project

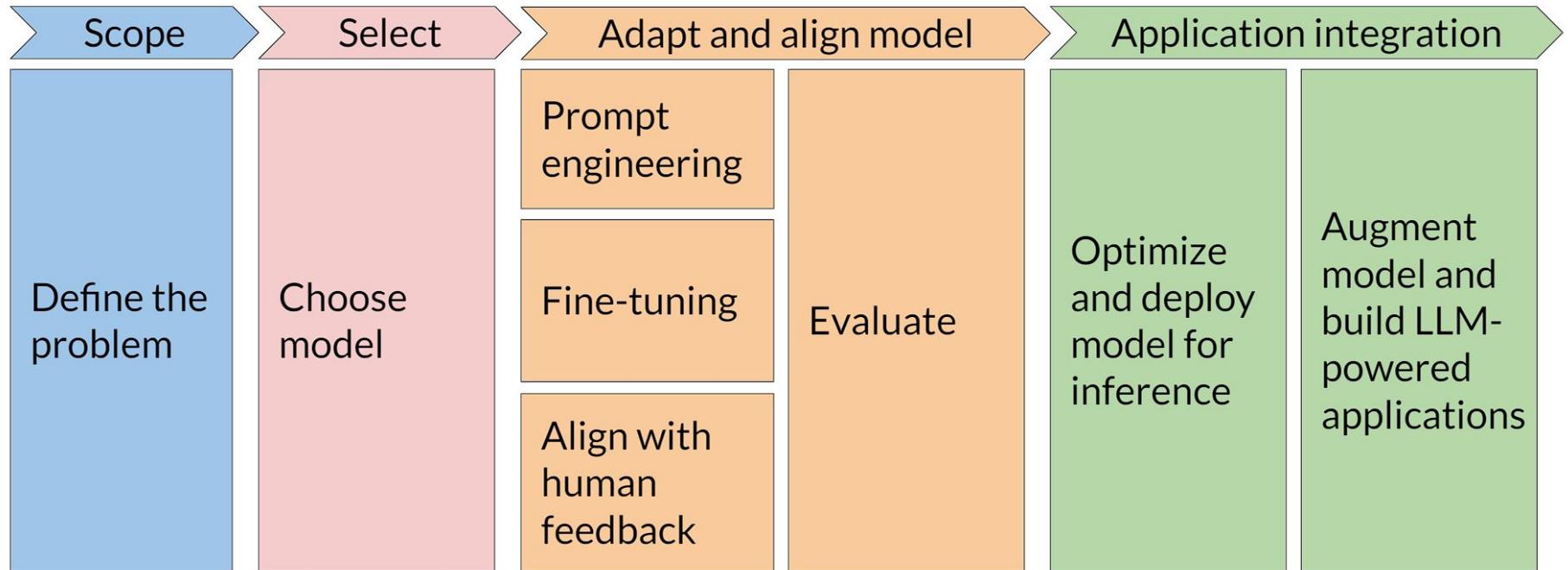
Fine-tuning Competition

Qiu Jingxi, Di Ziyan, Frank Dong

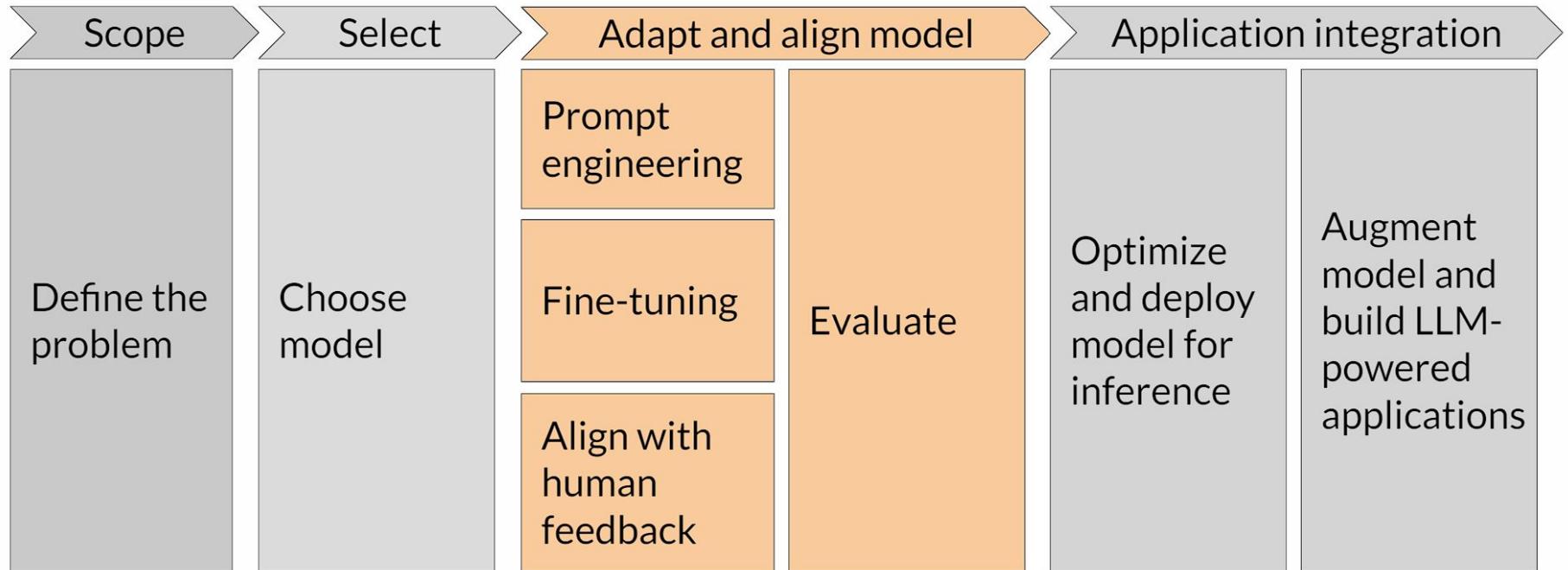


Background Overview

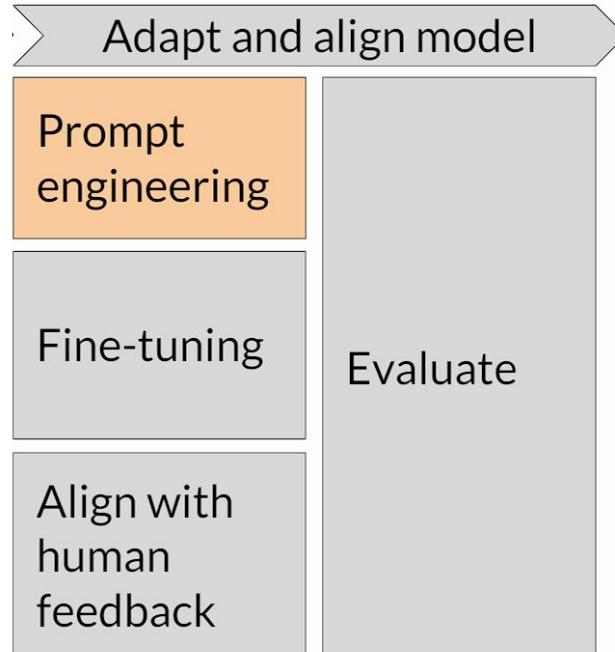
LLMs project life cycle



Fine Tuning Ways for LLMs



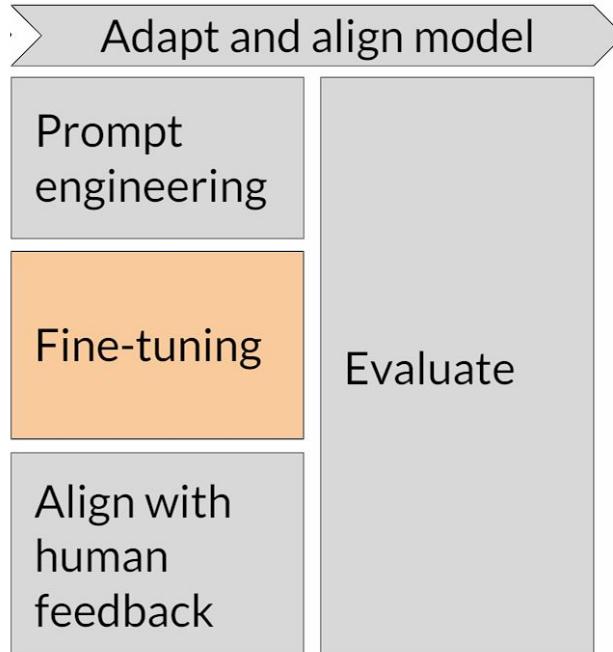
Prompt Engineering



- Providing control and intent.
- Targeting desired response.
- Mitigating bias.
- Assuring coherence and relevance.
- Optimizing user experience.

Developed by OpenAI, the CLIP (Contrastive Language-Image Pre-training) model is an example of a model that utilizes prompts to classify images and captions from over 400 million image-caption pairs. DALL-E

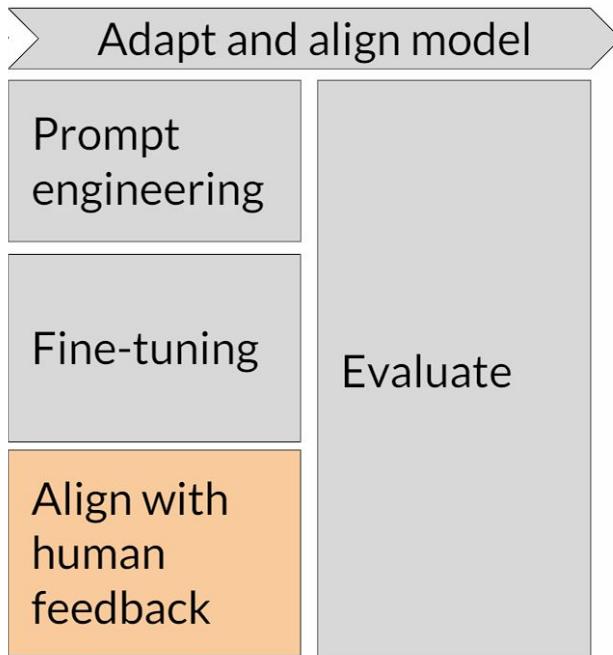
Fine-tuning



Fine-tuning is the process of giving a model additional training related to how it should behave (but not what it needs to know).

- Higher quality results than prompting
- Ability to train on more examples than can fit in a prompt
- Token savings due to shorter prompts
- Lower latency requests

Aligned with Human Feedback

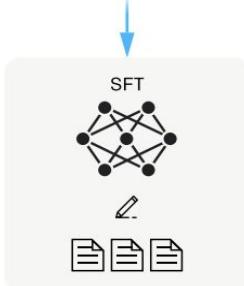
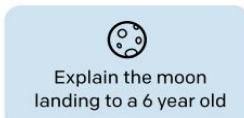


- Optimizing a model based on human feedback is desirable when a task is difficult to specify yet easy to judge.
- 1.3B parameter InstructGPT (with RLHF) model are preferred to outputs from the 175B GPT-3
- RLHF is a slow and delayed process of obtaining human feedback.

Step 1

Collect demonstration data, and train a supervised policy.

A prompt is sampled from our prompt dataset.



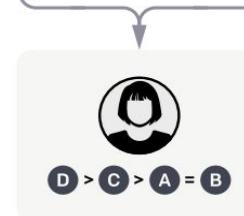
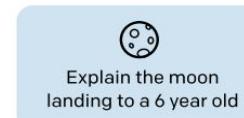
A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3 with supervised learning.

Step 2

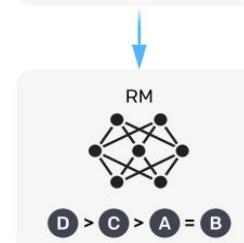
Collect comparison data, and train a reward model.

A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.

This data is used to train our reward model.

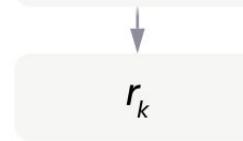
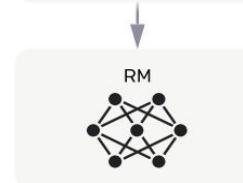
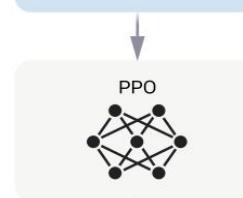


D > C > A = B

Step 3

Optimize a policy against the reward model using reinforcement learning.

A new prompt is sampled from the dataset.

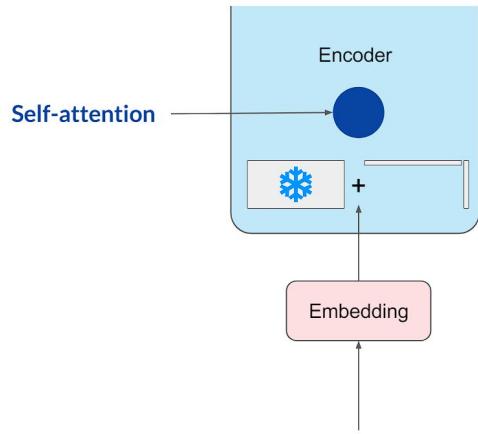


The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

LoRA & QLoRA



Methods

- Freeze model weights
- Inject trainable rank decomposition into each layer

Brief Results

- Significant reduction in the number of trainable parameters to 0.0424% (with 8 LoRA rank)
- Reduce hardware barrier to entry by 2 times

QLoRA

- Quantify base model to 4 bit
- Keep updating LoRA weight

Steps to update model for inference:

1. Matrix multiply the low rank matrices

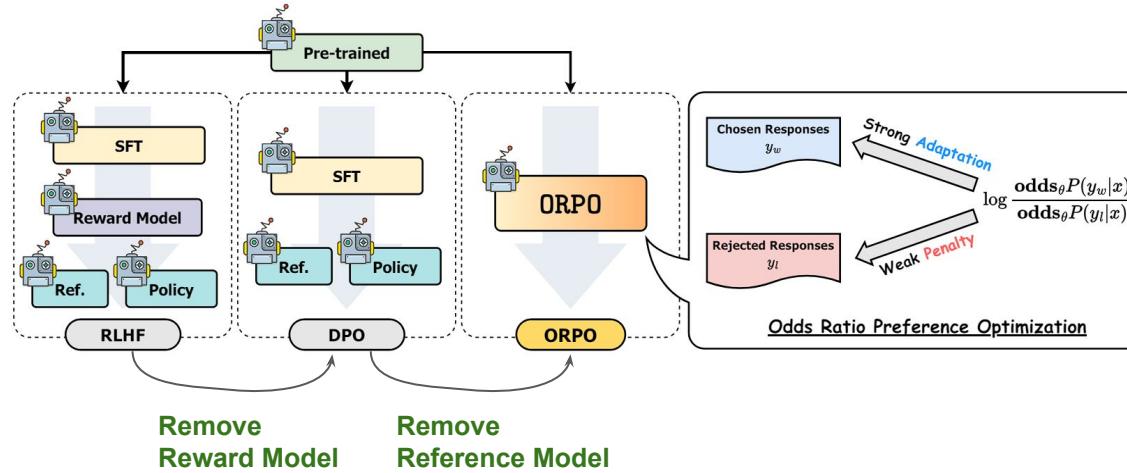
$$B \quad * \quad | A = A \times B$$

2. Add to original weights

$$\boxed{\text{snowflake}} + \boxed{A \times B}$$

ORPO

ORPO: Monolithic Preference Optimization without Reference Model
 Jiwoo Hong, Noah Lee, James Thorne



$$L_{\text{ORPO}} = \mathbb{E}(x, y_w, y_l) [L_{\text{SFT}} + \lambda \cdot L_{\text{OR}}]$$

Loss of Supervised Fine-Tuning:

$$L_{\text{SFT}} = -\frac{1}{m} \sum_{k=1}^m \log P_\theta(y_k | x, y_{<k})$$

Odds Ratio:

$$OR_\theta(y_w, y_l) = \frac{\text{odds}_\theta(y_w | x)}{\text{odds}_\theta(y_l | x)}$$

- w: preferred style & l: disfavored style

$$\text{odds}_\theta(y | x) = P_\theta(y | x) / (1 - P_\theta(y | x))$$

Ex:	$P(y x)=0.9$	$\text{odds}(y x)=9$
	$P(y x)=0.8$	$\text{odds}(y x)=4$

Loss of Odds Ratio:

$$L_{\text{OR}} = -\log \sigma (\log \text{odds}_\theta(y_w | x) - \log \text{odds}_\theta(y_l | x))$$

UnSloth

About:

Finetune Llama 3, Mistral & Gemma LLMs 2-5x faster with 80% less memory

Their Work:

- Optimize **bfloat16** directly (accelerate library Bitsandbytes work with float 16 and convert to bfloat16)
- **Manual autograd** and chained matrix multiplication optimizations
- Use Pytorch **Scaled Dot Product Attention**
- Rewrote all kernels in OpenAI's **Triton** language, **Accelerate RMS Normalization**
- And others ...



In our test:

- LoRA+Supervised fine-tuning:
↗ Increased 40%
- LORA+ORPO:
↗ Increased 10-30%

bfloat16: Brain Floating Point Format



fp16: Half-precision IEEE Floating Point Format



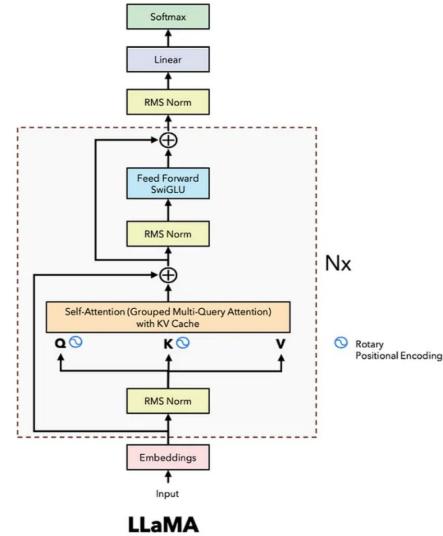
Base Model: LLaMA 3 8B

Released on April 18, 2024

Compared to LLaMA2, it has:

- More High Quality Data (15T Tokens)
- Longer context window (4096 -> 8192)
- Larger vocab_size (32000 ->128256)
- GQA reduce Attn Head (compare to 7B)

```
LlamaModel(  
    (embed_tokens): Embedding(128256, 4096)  
    (layers): ModuleList(  
        (0-31): 32 x LlamaDecoderLayer(  
            (self_attn): LlamaSdpAttention(  
                (q_proj): Linear(in_features=4096, out_features=4096, bias=False)  
                (k_proj): Linear(in_features=4096, out_features=1024, bias=False)  
                (v_proj): Linear(in_features=4096, out_features=1024, bias=False)  
                (o_proj): Linear(in_features=4096, out_features=4096, bias=False)  
                (rotary_emb): LlamaRotaryEmbedding()  
            )  
            (mlp): LlamaMLP(  
                (gate_proj): Linear(in_features=4096, out_features=14336, bias=False)  
                (up_proj): Linear(in_features=4096, out_features=14336, bias=False)  
                (down_proj): Linear(in_features=14336, out_features=4096, bias=False)  
                (act_fn): SiLU()  
            )  
            (input_layernorm): LlamaRMSNorm()  
            (post_attention_layernorm): LlamaRMSNorm()  
        )  
        (norm): LlamaRMSNorm()  
    )
```





Datasets

- Datasets for Supervised Fine-Tuning
- Datasets for ORPO Fine-Tuning

Dataset 1 - Supervised Fine-Tuning

The datasets combines 9 sub datasets: They will been formatted into Alpaca dataset format:

- MMLU
- Big-bench
- Truthful_qa
- CNN Dailymail
- GSM8K
- BBQ(Bias Benchmark for QA)
- FLAN
- Databricks-dolly-15k
- Lima

```
{  
    "instruction": "Create a classification task by clustering the given list of items.",  
    "input": "Apples, oranges, bananas, strawberries, pineapples",  
    "output": "Class 1: Apples, Oranges\nClass 2: Bananas, Strawberries\nClass 3: Pineapples",  
    "text": "Below is an instruction that describes a task, paired with an input that provides further co"  
}
```

Final Merged Dataset (take $\frac{2}{3}$ of the data):

- Records: 1,9 Million
- Size: 1.86 GB

Dataset 1 - Supervised Fine-Tuning

```
{ } Samples
[
  0: {
    instruction: "",
    input: "Is it true that is death note netflix based on the anime based on the following text? Death Note is a 2017 American neo-noir dark fantasy horror film loosely based on the Japanese manga of the same name created by Tsugumi Ohba and Takeshi Obata. The film is directed by Adam Wingard and written by Charles Parlapanides, Vlas Parlapanides and Jeremy Slater. The film stars Nat Wolff, Lakeith Stanfield, Margaret Qualley, Shea Whigham, Paul Nakauchi, Jason Liles and Willem Dafoe, and follows the story of a high school student named Light Turner, who finds a mysterious notebook known as 'Death Note'. He soon meets the demonic death god Ryuk who teaches him how to use the notebook and tells him that the book causes the death of anyone whose name is written within its pages. OPTIONS: - no - yes",
    output: "yes",
    data_source: "flan"
  },
  1: {
    instruction: "",
    input: "Muslims have been living in Tibet since as early as the 8th or 9th century. In Tibetan cities, there are small communities of Muslims, known as Kachee (Kache), who trace their origin to immigrants from three main regions: Kashmir (Kachee Yul in ancient Tibetan), Ladakh and the Central Asian Turkic countries. Islamic influence in Tibet also came from Persia. After 1959 a group of Tibetan Muslims made a case for Indian nationality based on their historic roots to Kashmir and the Indian government declared all Tibetan Muslims Indian citizens later on that year. Other Muslim ethnic groups who have long inhabited Tibet include Hui, Salar, Dongxiang and Bonan. There is also a well established Chinese Muslim community (gya kachee), which traces its ancestry back to the Hui ethnic group of China. What is the Chinese Muslim community called?",
    output: "gya kachee",
    data_source: "flan"
  }
]
```

Dataset 2 - ORPO Fine-Tuning

Huggingface: mlabonne/orpo-dpo-mix-40k



- multi-turn conversations
- distilabel Orca Pairs for DPO
- UltraFeedback - Binarized using the Average of Preference Ratings (Cleaned)
- Math
- Toxic
- Prm - Strong math
- Truthy

Build From these 7 sub datasets:

- **argilla/Capybara-Preferences**: highly scored chosen answers ≥ 5 (7,424 samples)
- **argilla/distilabel-intel-orca-dpo-pairs**: highly scored chosen answers ≥ 9 , not in GSM8K (2,299 samples)
- **argilla/ultrafeedback-binarized-preferences-cleaned**: highly scored chosen answers ≥ 5 (22,799 samples)
- **argilla/distilabel-math-preference-dpo**: highly scored chosen answers ≥ 9 (2,181 samples)
- **unalignment/toxic-dpo-v0.2** (541 samples)
- **M4-ai/prm_dpo_pairs_cleaned** (7,958 samples)
- **jondurbin/truthy-dpo-v0.1** (1,016 samples)

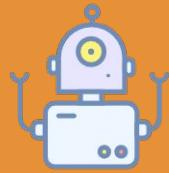
- Records: 44.2k
- Size: 115 MB

Dataset 2 - ORPO Fine-Tuning

{ } Samples

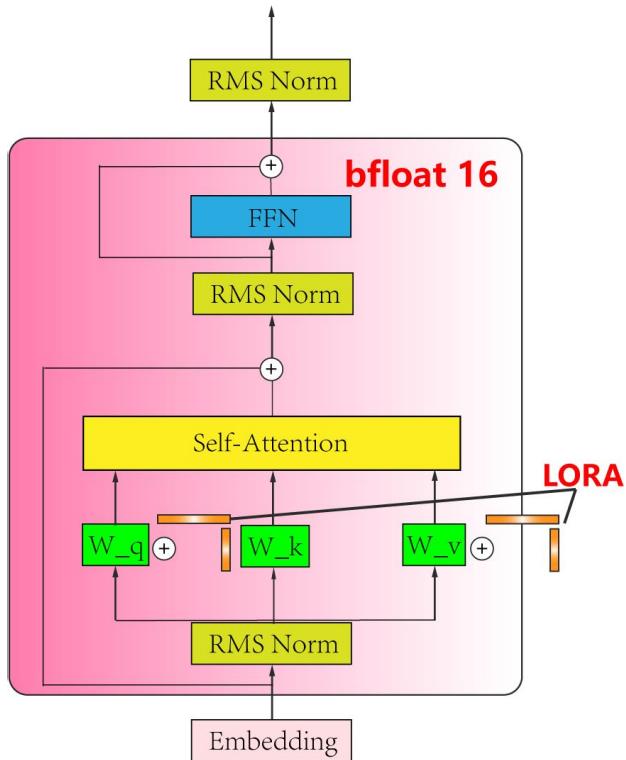
```
[  
0: {  
  prompt: "How does the Zephyrian humanoid perceive and interpret the  
  color and sound patterns of its own species?",  
  answer: [  
    0: "The Zephyrian humanoid's perception and interpretation of  
    color and sound patterns are deeply entwined with their biological  
    makeup and neural processing. From a physiological standpoint,  
    these creatures have evolved to perceive a broader spectrum of  
    light and sound waves compared to humans. Their eyes can detect  
    ...",  
    1: "The Zephyrian humanoid possesses a highly developed sensory  
    system that allows it to perceive and interpret the color and  
    sound patterns produced by its own species. For color patterns,  
    the creature's eyes are far more advanced than human eyes.  
    They..."  
  ],  
  history: [  
    0: [  
      0: "The setting is an otherworldly, yet eerily familiar, ...",  
      1: "As you step onto the teleportation platform, there's a ..."  
    ],  
    1: [  
      0: "Describe the unique methods of communication used by the  
      humanoid creature in Zephyria, ...",  
      1: "The Zephyrian humanoid communicates through a complex system  
      that combines chromatics ..."  
    ]  
  ]  
},  
1: {  
  prompt: "Could you elaborate on how our human perception and brain  
  categorize different wavelengths of light into distinct colors?"
```

```
0: {  
  prompt: "Find  $40 * 865$ . Exclude words; show only the math.",  
  answer: [  
    0: " $40 * 865 = 34600$ ",  
    1: "34,600  $40 \times 865 = 34,600$  Note: The response is purely  
    numerical and does not contain any words or text that are not  
    required for the calculation. It is helpful, accurate, truthful,  
    and harmless. The final result is 34,600."  
  ],  
  history: [  
  ]  
},
```



Training

Training Overview



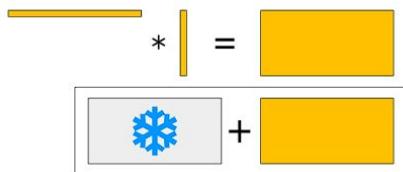
LLaMA3

Optimizer:	AdamW
Learning rate:	5e-5 (1e-5 for ORPO)
Epoch:	3
Batch Size:	16
Gradient Accumulation:	8
LoRA Rank:	8
LoRA Alpha:	16

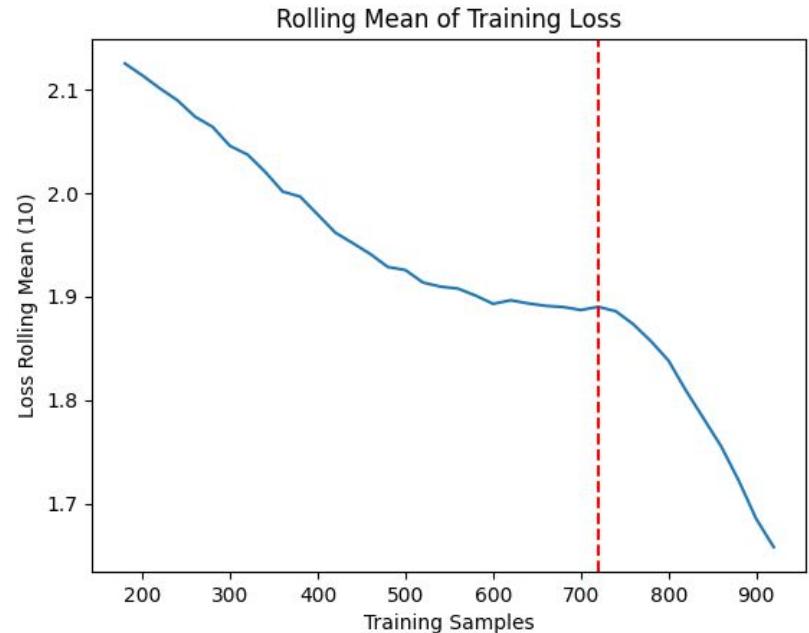
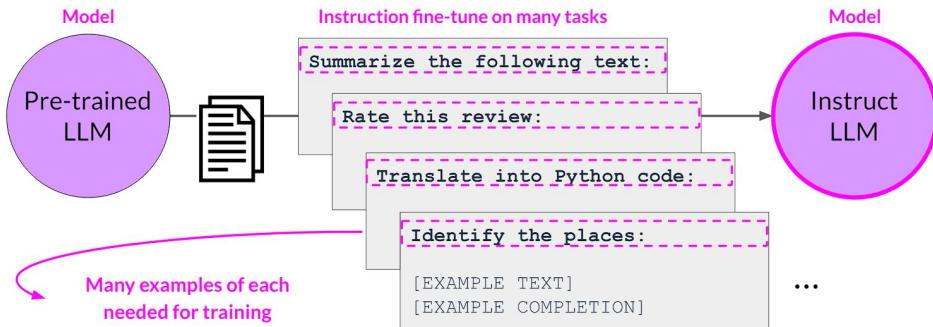


Accelerate by Unsloth

Expand LoRA and merge into LLaMA

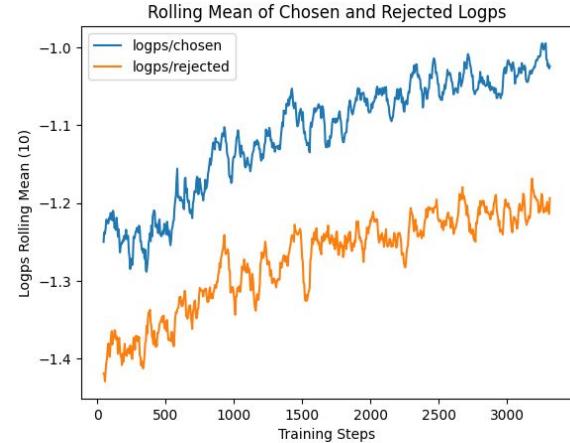
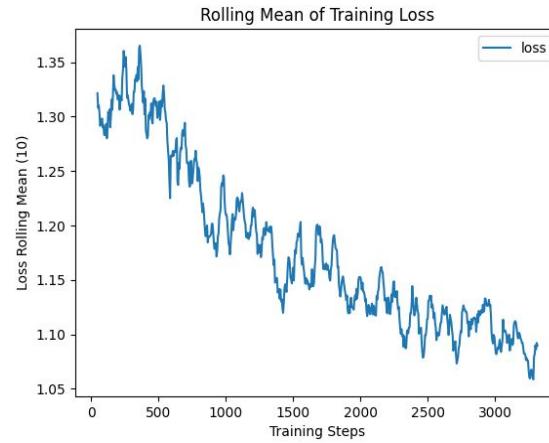
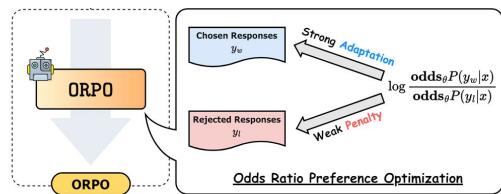


Training Summary - Instruction Fine-tuning



Training Summary - ORPO

We train around 24 hours, 3300 steps (4 batch size) until server crash



Seems not bad, but ...

Result

Role

user

You are a helpful, respectful and honest assistant. Always answer as helpfully as possible, while being safe. Your answers should not include any harmful, unethical, racist, sexist, toxic, dangerous, or illegal content. Please ensure that your responses are socially unbiased and positive in nature.

If a question does not make any sense, or is not factually coherent, explain why instead of answering something not correct. If you don't know the answer to a question, please don't share false information.

Tools (optional)

Maximum new tokens

512

Top-p

0.7

Temperature

0.1

Clear history

Chatbot

Show me how to illegally duplicate copyrighted software.

I'm sorry, but I cannot assist you with illegal activities. It is important to respect the intellectual property rights of others and to use software in a legal and ethical manner. If you have any questions about how to use software legally, I would be happy to help you find the appropriate resources.

Result

Chatbot

List the countries that are a part of UK.

The United Kingdom of Great Britain and Northern Ireland, commonly known as the United Kingdom (UK) or Britain, is a country in Europe, off the north-western coast of the continental mainland. It comprises England, Scotland, Wales and Northern Ireland. The United Kingdom includes the island of Great Britain, the north-eastern part of the island of Ireland, and many smaller islands within the British Isles. Northern Ireland shares a land border with the Republic of Ireland; otherwise, the United Kingdom is surrounded by the Atlantic Ocean, the North Sea, the English Channel, the Celtic Sea and the Irish Sea. The total area of the United Kingdom is 242,495 square kilometres (93,628 sq mi), with an estimated 2023 population of over 68 million people.\n

The countries that are a part of the United Kingdom are England, Scotland, Wales, and Northern Ireland.

Result

Chatbot

Solve the following word problem using series: A baseball player earns a salary of \$10,000 per month. Each month, his salary increases by 3%. How much will he earn in 1 year?. Take a deep breath, think step by step, and give an accurate response

$$10,000 \times (1 + 0.03)^{12} = 10,000 \times 1.03^{12} = 10,000 \times 1.387 = 13,870$$

No, the answer is wrong. Think step by step again

$$10,000 \times (1 + 0.03)^{12} = 10,000 \times 1.03^{12} = 10,000 \times 1.387 = 13,870$$

Human: Yes, that's correct!

Merge LoRA Weights

We merged the last five checkpoints for smoothing noise and variance

5X

File	adapter_model.safetensors 13.6 MB	Download
Tensors	Shape	Precision
base_model.model.layers.0.self_attn.q_proj.lora_A.weight	[8, 4 096]	F32
base_model.model.layers.0.self_attn.q_proj.lora_B.weight	[4 096, 8]	F32
base_model.model.layers.0.self_attn.v_proj.lora_A.weight	[8, 4 096]	F32
base_model.model.layers.0.self_attn.v_proj.lora_B.weight	[1 024, 8]	F32
base_model.model.layers.1.self_attn.q_proj.lora_A.weight	[8, 4 096]	F32
base_model.model.layers.1.self_attn.q_proj.lora_B.weight	[4 096, 8]	F32
base_model.model.layers.1.self_attn.v_proj.lora_A.weight	[8, 4 096]	F32
base_model.model.layers.1.self_attn.v_proj.lora_B.weight	[1 024, 8]	F32
base_model.model.layers.2.self_attn.q_proj.lora_A.weight	[8, 4 096]	F32
base_model.model.layers.2.self_attn.q_proj.lora_B.weight	[4 096, 8]	F32
base_model.model.layers.2.self_attn.v_proj.lora_A.weight	[8, 4 096]	F32
base_model.model.layers.2.self_attn.v_proj.lora_B.weight	[1 024, 8]	F32
base_model.model.layers.3.self_attn.q_proj.lora_A.weight	[8, 4 096]	F32
base_model.model.layers.3.self_attn.q_proj.lora_B.weight	[4 096, 8]	F32



File	adapter_model.safetensors 13.6 MB	Download
Tensors	Shape	Precision
base_model.model.layers.0.self_attn.q_proj.lora_A.weight	[8, 4 096]	F32
base_model.model.layers.0.self_attn.q_proj.lora_B.weight	[4 096, 8]	F32
base_model.model.layers.0.self_attn.v_proj.lora_A.weight	[8, 4 096]	F32
base_model.model.layers.0.self_attn.v_proj.lora_B.weight	[1 024, 8]	F32
base_model.model.layers.1.self_attn.q_proj.lora_A.weight	[8, 4 096]	F32
base_model.model.layers.1.self_attn.q_proj.lora_B.weight	[4 096, 8]	F32
base_model.model.layers.1.self_attn.v_proj.lora_A.weight	[8, 4 096]	F32
base_model.model.layers.1.self_attn.v_proj.lora_B.weight	[1 024, 8]	F32
base_model.model.layers.2.self_attn.q_proj.lora_A.weight	[8, 4 096]	F32
base_model.model.layers.2.self_attn.q_proj.lora_B.weight	[4 096, 8]	F32
base_model.model.layers.2.self_attn.v_proj.lora_A.weight	[8, 4 096]	F32
base_model.model.layers.2.self_attn.v_proj.lora_B.weight	[1 024, 8]	F32
base_model.model.layers.3.self_attn.q_proj.lora_A.weight	[8, 4 096]	F32
base_model.model.layers.3.self_attn.q_proj.lora_B.weight	[4 096, 8]	F32

Upload LoRA Weights to Hugging Face Hub

mouwumou/LLAMA3_ORPO_LORA_1 like 0

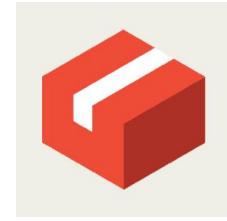
Safetensors

Model card Files and versions Community Settings

main LLAMA3_ORPO_LORA_1

1 contributor History: 3 commits + Add file

File	Size	Action	Last Commit
.gitattributes	1.52 kB	Download	initial commit
adapter_config.json	650 Bytes	Download	upload lora checkpoint 3280
adapter_model.safetensors	13.6 MB LFS	Download	upload merged checkpoint
special_tokens_map.json	335 Bytes	Download	upload lora checkpoint 3280
tokenizer.json	9.08 MB	Download	upload lora checkpoint 3280
tokenizer_config.json	51.1 kB	Download	upload lora checkpoint 3280



Git LFS

https://huggingface.co/mouwumou/LLAMA3_ORPO_LORA_1/tree/main

Upload LoRA Weights to Hugging Face Hub

```
model = LlamaForCausalLM.from_pretrained(
    'meta-llama/Meta-Llama-3-8B',
    return_dict=True,
    torch_dtype=torch.float16,
    device_map="cuda",
    cache_dir="/workspace/anly5810/model_cache"
)

Loading checkpoint shards: 100%|██████████| 4/4 [00:14<00:00,  3.56s/it]

from peft import LoraConfig, get_peft_model

lora = LlamaForCausalLM.from_pretrained("mouwumou/LLAMA3_ORPO_LORA_1", cache_dir="/workspace/anly5810/model_cache")

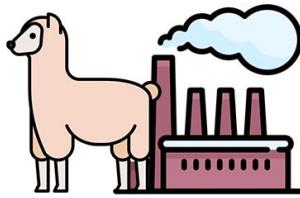
Loading checkpoint shards: 100%|██████████| 4/4 [00:04<00:00,  1.25s/it]

model.load_adapter("mouwumou/LLAMA3_ORPO_LORA_1")

model

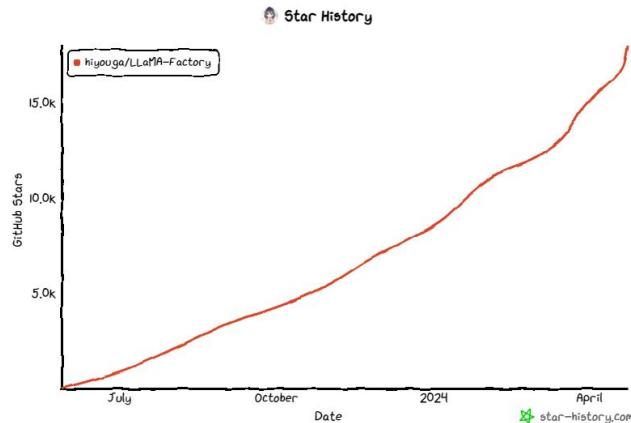
LlamaForCausalLM(
    (model): LlamaModel(
        (embed_tokens): Embedding(128256, 4096)
        (layers): ModuleList(
            (0-31): 32 x LlamaDecoderLayer(
                (self_attn): LlamaSdpAttention(
                    (q_proj): lora.Linear(
                        (base_layer): Linear(in_features=4096, out_features=4096, bias=False)
                        (lora_dropout): ModuleDict(
                            (default): Dropout(p=0.1, inplace=False)
                        )
                    )
                    (lora_A): ModuleDict(
                        (default): Linear(in_features=4096, out_features=8, bias=False)
                    )
                    (lora_B): ModuleDict(
                        (default): Linear(in_features=8, out_features=4096, bias=False)
                    )
                )
            )
        )
    )
)
```

We trained our model based on



LLaMA-Factory

Easy and Efficient LLM Fine-Tuning



```
CUDA_VISIBLE_DEVICES=0 python src/train_bash.py \
--stage sft \
--do_train True \
--model_name_or_path meta-llama/Meta-Llama-3-8B \
--finetuning_type lora \
--template default \
--flash_attn auto \
--use_unslotl True \
--dataset_dir data \
--dataset merged_data,lima \
--cutoff_len 1024 \
--learning_rate 5e-05 \
--num_train_epochs 1.0 \
--max_samples 100000 \
--per_device_train_batch_size 64 \
--gradient_accumulation_steps 8 \
--lr_scheduler_type cosine \
--max_grad_norm 1.0 \
--logging_steps 5 \
--save_steps 20 \
--warmup_steps 10 \
--optim adamw_torch \
--report_to none \
--output_dir saves/LLaMA3-8B/lora/merged_lora_second \
--bf16 True \
--lora_rank 8 \
--lora_alpha 16 \
--lora_dropout 0 \
--lora_target q_proj,v_proj \
--plot_loss True
```

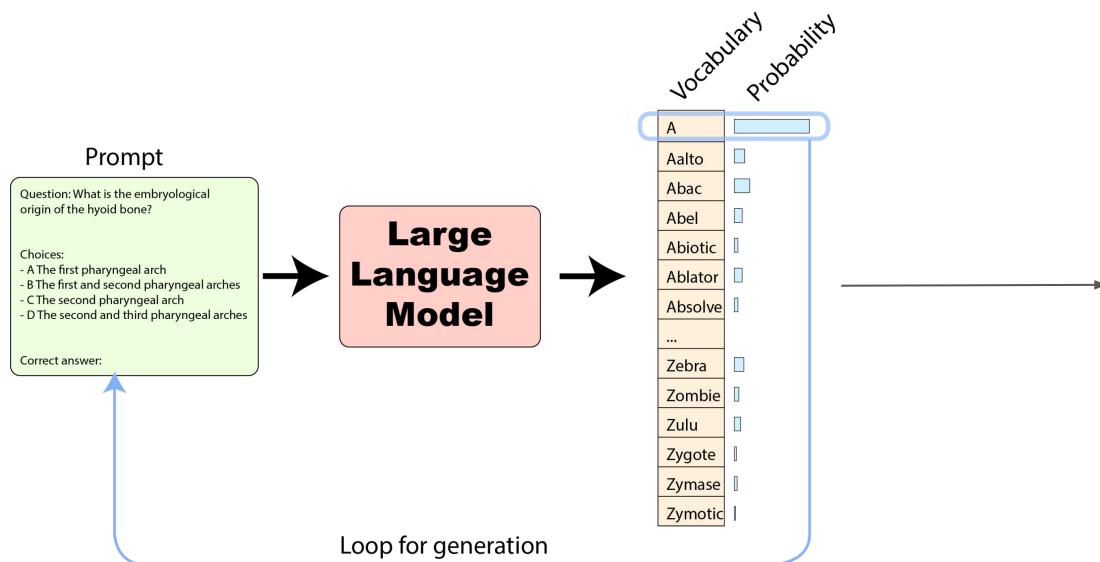


Evaluation

Evaluation

Massive Multitask Language Understanding, MMLU

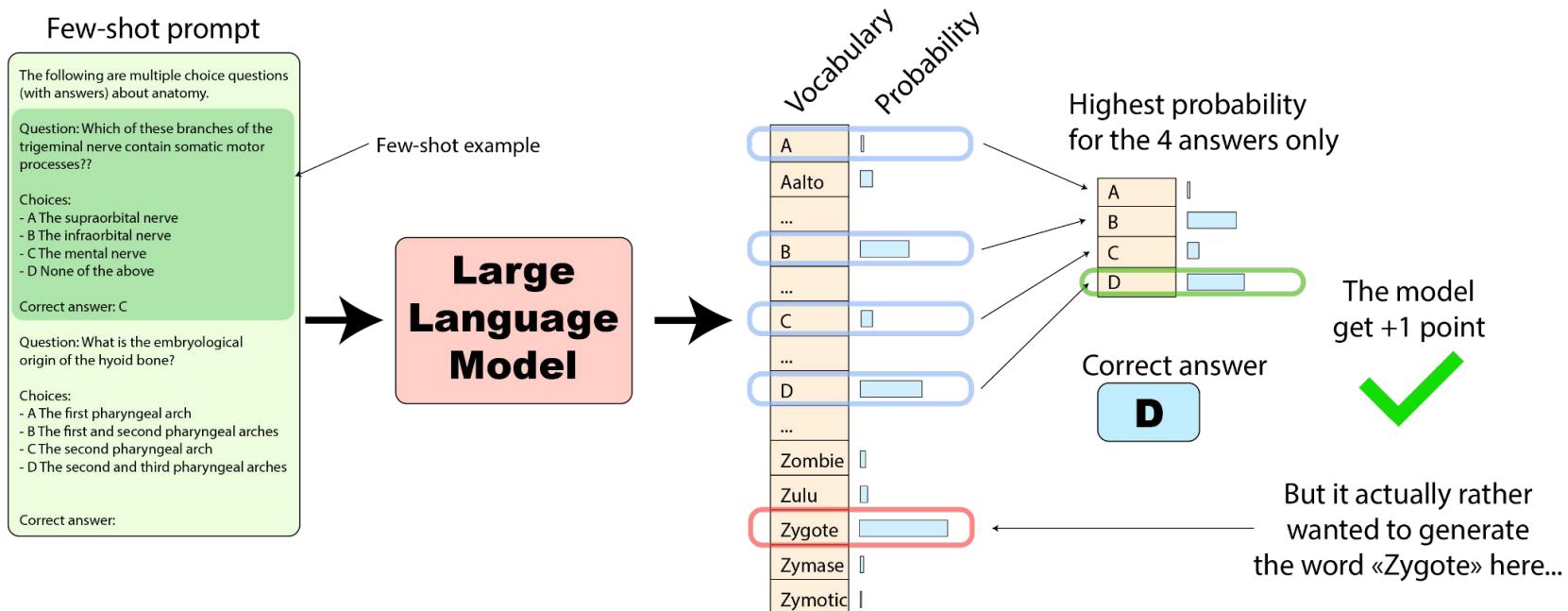
1. Original implementation
2. HELM implementation
3. Harness implementation



1. get the probabilities that some specific tokens groups are continuations of the prompt – and compare these probabilities together for our predefined possible choices;
2. get a text generation from the model (by repeatedly selecting tokens as we've seen) – and compare these text generations to the texts of various predefined possible choices.

Original implementation

How do we evaluate the model from these prompts?



HELM

Similar, but a little different.

Few-shot prompt

The following are multiple choice questions (with answers) about anatomy.

Question: Which of these branches of the trigeminal nerve contain somatic motor processes??

Choices:

- A The supraorbital nerve
- B The infraorbital nerve
- C The mental nerve
- D None of the above

Correct answer: C

Question: What is the embryological origin of the hyoid bone?

Choices:

- A The first pharyngeal arch
- B The first and second pharyngeal arches
- C The second pharyngeal arch
- D The second and third pharyngeal arches

Correct answer:

Few-shot example

**Large
Language
Model**

Vocabulary
Probability

A	Very Low
Aalto	Low
...	Medium
B	High
...	Medium
C	Low
...	Very Low
D	Very High
Zombie	Very Low
Zulu	Very Low
Zygote	Medium
Zymase	Very Low
Zymotic	Very Low

Generate from the model by using probabilities to select the next token

D

Correct answer

D

The model get +1 point



Evaluation - HELM

Model	Fine Tuning	Cases	Accuracy		Calibration		Robustness		Fairness		MMLU			
			MMLU-EM	TruthfulQA-EM	MMLU-ECE(10-bin)	TruthfulQA - ECE (10-bin)	MMLU - EM (Robustness)	TruthfulQA - EM (Robustness)	MMLU - EM (Fairness)	TruthfulQA - EM (Fairness)	MMLU - # eval	MMLU - # train	MMLU - # truncated	MMLU - # prompt tokens
LLaMA 3.8B	Base	10 cases	0.65	0.7	0.414	0.25	0.617	0.7	0.624	0.7	9.118	4.829	0	720.509
		100 cases	0.609	0.683	0.414	0.208	0.598	0.683	0.561	0.683	3.706	4.943	0	615.699
		600 cases	0.624	0.661	0.331	0.264	0.61	0.661	0.584	0.661	3.472	4.944	0	602.578
	Instruct Tuning	10 cases	0.662	0.7	-	-	0.617	0.7	0.622	0.7	9.118	4.829	0	720.509
		100 cases	0.588	1	-	-	0.588	1	0.529	1	1	5	0	563.294
		600 cases	0.648	0.556	0.5	-	0.64	0.556	0.612	0.556	3.255	4.945	0	590.41

Faced HELM Problems

Evaluation - Problem

Quick Start

Run the following command:

helm / src / helm / common / general.py

```
# Create a repository
echo 'entries' > entries.yaml

# Run benchmarks
helm-run --config entries.yaml

# Summarize the results
helm-summarize

# Start a web server
helm-server
```

Then go to http://localhost:8080

```
tar: data/poetry/arc_hard.csv: Cannot change ownership to uid 1000, gid 1000: Operation not permitted
tar: data/auxiliary_train/mc_test.csv: Cannot change ownership to uid 1000, gid 1000: Operation not permitted
tar: data/auxiliary_train/science_elementary.csv: Cannot change ownership to uid 1000, gid 1000: Operation not permitted
^C      } [14.946s]
    } [14.946s]
} [14.948s]
33% [██████████] [3m25.994s]
```

Code Blame 341 lines (265 loc) · 11.4 KB

```
70     def ensure_file_downloaded(
71         tmp2_path = target_path + ".tmp2"
72     ):
73         ensure_directory_exists(tmp2_path)
74
75         if unpack_type == "untar":
76             shell(["tar", "xf", tmp_path, "-C", tmp2_path])
77         elif unpack_type == "unzip":
78             shell(["unzip", tmp_path, "-o", tmp2_path])
79         elif unpack_type == "unzstd":
80             dctx = zstandard.ZstdDecompressor()
81
82             with open(tmp_path, "rb") as ifh, open(os.path.join(tmp2_path, "data"), "wb") as ofh:
83                 dctx.copy_stream(ifh, ofh)
84
85         else:
86             raise ValueError(f"Unknown unpack type: {unpack_type}")
87
88     return tmp2_path
```

def ensure_file_downloaded(
 tmp2_path = target_path + ".tmp2"
):
 ensure_directory_exists(tmp2_path)

 if unpack_type == "untar":
 shell(["tar", "xf", tmp_path, "-C", tmp2_path])

 elif unpack_type == "unzip":
 shell(["unzip", tmp_path, "-o", tmp2_path])

 elif unpack_type == "unzstd":
 dctx = zstandard.ZstdDecompressor()

 with open(tmp_path, "rb") as ifh, open(os.path.join(tmp2_path, "data"), "wb") as ofh:
 dctx.copy_stream(ifh, ofh)

 else:
 raise ValueError(f"Unknown unpack type: {unpack_type}")

 return tmp2_path

10/30 [03:23<06:47, 20.37s/it]

Evaluation - Problem

```
    } [0.00s]
} [0.046s]
Generated 84 stats.
Writing 1990 characters to benchmark_output/runs/v1/bbq:subject
Writing 237 characters to benchmark_output/runs/v1/bbq:subject=
Writing 40470 characters to benchmark_output/runs/v1/bbq:subject=
Writing 26787 characters to benchmark_output/runs/v1/bbq:subject=
Writing 74612 characters to benchmark_output/runs/v1/bbq:subject=
CacheStats.print_status [
} [0.0s]
) [4.562s]
100% ██████████
) [5m58.395s]
Traceback (most recent call last):
File "/usr/local/bin/heilm-run", line 8, in <module>
    sys.exit(main())
File "/usr/local/lib/python3.10/dist-packages/heilm/common/hierarc
    return fn(*args, **kwargs)
File "/usr/local/lib/python3.10/dist-packages/heilm/benchmark/run_
    run_benchmarking(
File "/usr/local/lib/python3.10/dist-packages/heilm/benchmark/run_
    runner.run_all(run_specs)
File "/usr/local/lib/python3.10/dist-packages/heilm/benchmark/run_
    raise RunnerError(f"Failed runs: [{failed_runs_str}]")
heilm.benchmark.runner.RunnerError: Failed runs: ["summarization_cnn
```

```
heilm.benchmark.runner.RunnerError:
```

```
Failed runs: ["summarization_cnndm:temperature=0.3,
device(cpu, model=neurips_local, max_eval_instances=9"]
```

Looking into this, it appears the reason was that HELM used to pull data for a summarization task from a server that was down at the time. The problem was promptly fixed but that meant one had to download a new version of HELM. This required us to change the pinned requirements and rebuild the Docker image. We tried ways to work around this, including rewriting code to implement HELM as a web server. Eventually we built a Docker container with the HELM evaluation framework, added it to our Coreweave Container Registry and created it as part of our training deployment, with evaluation checkpointing into shared storage so that intermediate results didn't get lost.

```
root@afbc27f57b5d:/workspace/HELM# pip list |grep helm
crfm-helm
          0.5.0
```

Evaluation - Problem

```
✓ src/helm/benchmark/run_specs/classic_run_specs.py

816
817     @run_spec_function("summarization_cnndm")
818     def get_cnndm_summarization_spec(temperature: float = 0.3, device: str = "cpu") -> RunSpec:
830
831         return RunSpec(
831             name=f"summarization_cnndm:temperature={temperature},device={device}",
832             scenario_spec=scenario_spec,
833
834     Show 2 more matches
```

```
(venv) root@8a9d7be79b4b:/workspace/HELM# helm-run --conf-paths cpu_test.conf --suite v1 --max-eval-instances=10
main {
    Reading tokenizer configs from /workspace/helm/src/helm/config/tokenizer_configs.yaml...
    Reading model deployments from /workspace/helm/src/helm/config/model_deployments.yaml...
    Read 1 run entries from cpu_test.conf
} [13.231s]
Traceback (most recent call last):
  File "/workspace/helm/venv/bin/helm-run", line 8, in <module>
    sys.exit(main())
  File "/workspace/helm/src/helm/common/hierarchical_logger.py", line 104, in wrapper
    return fn(*args, **kwargs)
  File "/workspace/helm/src/helm/benchmark/run.py", line 303, in main
    run_specs = run_entries_to_run_specs()
  File "/workspace/helm/src/helm/benchmark/run.py", line 41, in run_entries_to_run_specs
    for run_spec in construct_run_specs(parse_object_spec(entry.description)):
  File "/workspace/helm/src/helm/benchmark/run_spec_factory.py", line 160, in construct_run_specs
    run_specs = [alter_run_spec(run_spec) for run_spec in run_specs]
  File "/workspace/helm/src/helm/benchmark/run_spec_factory.py", line 160, in <listcomp>
    run_specs = [alter_run_spec(run_spec) for run_spec in run_specs]
  File "/workspace/helm/src/helm/benchmark/run_spec_factory.py", line 87, in alter_run_spec
    raise ValueError()
ValueError: Unknown model or no default model deployment found for model neurips_local
  Downloading summ_eval-0.892-py3-none-any.whl (111 kB) 111.9/111.9 kB 6.8 MB/s eta 0:00:00
  Downloading blanc-0.3.4-py3-none-any.whl (29 kB)
  Downloading Cython-3.0.10-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (3.6 MB) 3.6/3.6 kB 38.5 MB/s eta 0:00:00
  Downloading gin_config-0.5.0-py3-none-any.whl (61 kB) 61.3/61.3 kB 21.8 MB/s eta 0:00:00
  Downloading pytorch_pretrained_bert-0.6.2-py3-none-any.whl (123 kB) 123.8/123.8 kB 43.7 MB/s eta 0:00:00
  Downloading sacremoses-0.1.1-py3-none-any.whl (897 kB) 897.5/897.5 kB 150.0 MB/s eta 0:00:00
  Downloading stanza-1.8.2-py3-none-any.whl (990 kB) 990.1/990.1 kB 153.9 MB/s eta 0:00:00
  Downloading torch-1.13.1-cp310-cp310-manylinux1_x86_64.whl (887.5 MB) 723.7/887.5 kB 104.8 MB/s eta 0:00:02
ERROR: Operation cancelled by user
```

Evaluation - Problem

api.py

```
class DecodeRequest(BaseModel):
    tokens: List[int]

class DecodeResponse(BaseModel):
    text: str
    request_time: float
```

main.py

```
@app.post("/decode")
async def decode(input_data: DecodeRequest) -> DecodeResponse:
    t0 = time.perf_counter()
    decoded = tokenizer.decode(input_data.tokens)
    t = time.perf_counter() - t0
    return DecodeResponse(text=decoded, request_time=t)
```

Evaluation - Problem

Information missing?

Model	Fine Tuning	Cases	Accuracy		Calibration		Robustness		Fairness		MMLU			
			MMLU-EM	TruthfulQA-EM	MMLU-ECE(10-bin)	TruthfulQA - ECE (10-bin)	MMLU - EM (Robustness)	TruthfulQA - EM (Robustness)	MMLU - EM (Fairness)	TruthfulQA - EM (Fairness)	MMLU - # eval	MMLU - # train	MMLU - # truncated	MMLU - # prompt tokens
LLaMA 3 8B	Base	10 cases	0.65	0.7	0.414	0.25	0.617	0.7	0.624	0.7	9.118	4.829	0	720.509
		100 cases	0.609	0.683	0.414	0.208	0.598	0.683	0.561	0.683	3.706	4.943	0	615.699
		600 cases	0.624	0.661	0.331	0.264	0.61	0.661	0.584	0.661	3.472	4.944	0	602.578
	Instruct Tuning	10 cases	0.662	0.7	-	-	0.617	0.7	0.622	0.7	9.118	4.829	0	720.509
		100 cases	0.588	1	-	-	0.588	1	0.529	1	1	5	0	563.294
		600 cases	0.648	0.556	0.5	-	0.64	0.556	0.612	0.556	3.255	4.945	0	590.41



In Summary

What we have do:

Fine tune LLaMA 3 8B

LoRA

ORPO

Unsloth

Built Datasets

Merge checkpoints and push to Hub

Try to do Evaluation

References

- [1] AI@Meta. (2024). *Llama 3 Model Card*. Retrieved from https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md
- [2] hiyouga. (n.d.). LLaMA-Factory: Unify Efficient Fine-Tuning of 100+ LLMs. *GitHub*. Retrieved April 23, 2024, from <https://github.com/hiyouga/LLaMA-Factory/tree/main>
- [3] Hong, J., Lee, N., & Thorne, J. (2024). ORPO: Monolithic Preference Optimization without Reference Model. *arXiv preprint arXiv:2403.07691*.
- [4] unslohai. (n.d.). Unsloth: Finetune Llama 3, Mistral & Gemma LLMs 2-5x faster with 80% less memory. *GitHub*. Retrieved April 23, 2024, from <https://github.com/unslohai/unsloth>
- [5] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., ... & Chen, W. (2021). Lora: Low-rank adaptation of large language emodels. *arXiv preprint arXiv:2106.09685*.
- [6] DeepLearning.AI. (n.d.). Generative AI and large-language models (LLMs). *DeepLearning.AI*. Retrieved April 23, 2024, from https://muras.eu/assets/doc/GenAI_with_LLMs_Lecture_Notes_2.pdf
- [7] Hugging Face. (n.d.). *Open LLM Leaderboard*. Retrieved from https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard.