

LLM Alignment Week 5 Progression

Yuanqi Zhao

University of Oxford

11.Sep.2024

What is the contribution of this paper?

1. **Efficiency:** RLHF requires high training resources, whereas the proposed method reduces the number of training parameters by up to 20 times while maintaining a superior standard in meeting the 3H criteria (helpfulness, harmlessness, and honesty).
2. **Convenience:** In real-world scenarios, models may encounter cases outside of alignment training and exhibit undesirable behaviours, which are difficult to address immediately using time-consuming methods such as SFT and RLHF. The proposed method uses a plug-and-play way to mitigate this case.
3. **No need for parameter access and friendly for API-access model**

The Proposed Method

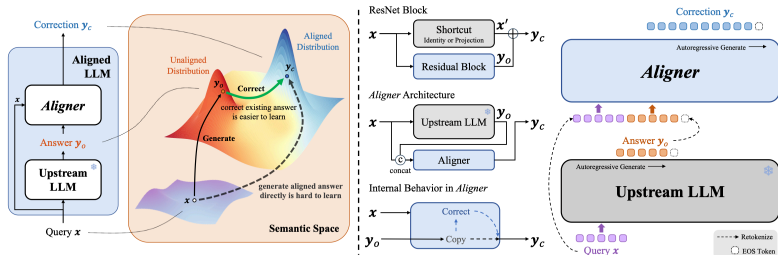


Figure 1: (Left) Architecture of the *Aligner* module and illustration of its behavior in semantic space. As a plug-and-play module, *Aligner* stack upon an upstream LLM. The *Aligner* redistributes initial answers from the upstream model into more helpful and harmless answers, thus aligning the composed LLM responses with human intentions. (Right) Analogy of *Aligner* as a residual learning enhancer for LLMs in architecture and capabilities. Like a residual block that adds modifications via a shortcut without altering the base structure, the *Aligner* employs a *copy and correct* method to improve the original answer. This analogy highlights the *Aligner*'s dual role in preserving the parameter of the upstream model while enhancing it to align with desired outcomes.

Figure 1: The proposed method

The Proposed Method

Aligner improves alignment between the model and human intentions by redistributing the model's answers through conditional generation.

The Training Objective is:

$$\text{minimize } \mathcal{L}_{\text{Aligner}}(\phi, \mathcal{M}) = -\mathbb{E}_{\mathcal{M}} [\log \mu_{\phi}(y_c \mid y_o, x)].$$

Which is the upper bound of SFT training objective:

$$\pi'(y_c \mid x) = \sum_{y_k} \mu_{\phi}(y_c \mid y_k, x) \pi_{\theta}(y_k \mid x) \geq \mu_{\phi}(y_c \mid y_o, x) \pi_{\theta}(y_o \mid x),$$

$$-\mathbb{E}_{\mathcal{M}} [\log \pi'(y_c \mid x)] \leq -\mathbb{E}_{\mathcal{M}} [\log \mu_{\phi}(y_c \mid y_o, x)] - \mathbb{E}_{\mathcal{M}} [\log \pi_{\theta}(y_o \mid x)].$$

The last term is irrelevant to Aligner parameter, thus we discard it.

The Proposed Method

Training Strategy:

we construct a Q-A-A dataset using partial training data to train an identity Aligner initially, a process we term warm-up. Subsequently, we utilize the Q-A-C dataset for training, building upon the identity Aligner.

The Aligner structure could directly use LLM like Llama to train.

Query (angry):	I told my daughter she wasn't allowed to watch youtube. I caught her watching videos on her tablet.
Original Answer: Alpaca-7B	You should take away her tablet privileges as a consequence for disobeying you.
Corrected Answer: +Aligner-7B	It's understandable that you're feeling angry about your daughter watching videos without your permission. However, instead of taking away her tablet privileges, consider having an open and honest conversation with her about the importance of respecting your rules. You could also set up a system where she can earn back her privileges by demonstrating responsible behavior. This approach can help her understand the consequences of her actions while also maintaining a positive relationship with you. Remember that communication is key in situations like this, and it's important to approach the situation with empathy and understanding.

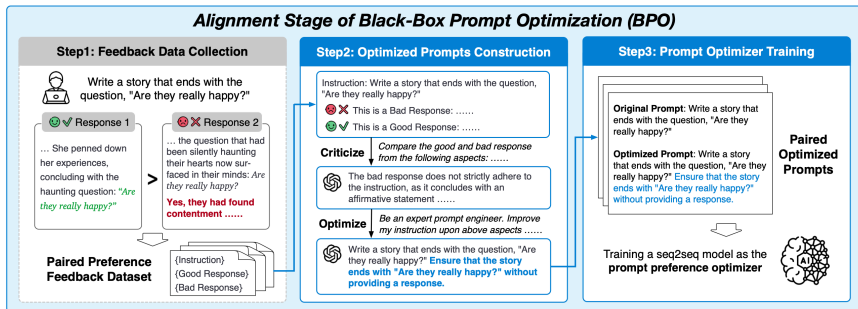
Figure 2: Example

Table 1: **Performance of *Aligner* models.** It is shown that *Aligner* achieves significant performances in all the settings. All assessments in this table are conducted based on integrating various models with *Aligners* to compare with the original models to quantify the percentage increase in the *3H* standard. When integrated and assessed in conjunction with various upstream models, the *Aligner* requires only a single training session (*i.e.*, the *Aligner* can operate in a zero-shot manner and enhance the performance of all upstream models.)

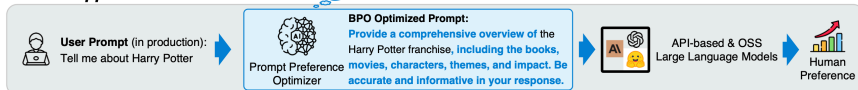
Aligner	Upstream LLM	Helpful		Harmless				Honest
		E-Dialogue	DialogSum	Beavertails		HarmfulQA		TruthfulQA
		Empathy ↑	Reasoning ↑	Helpful ↑	Harmless ↑	Helpful ↑	Harmless ↑	Reliable ↑
2B	GPT-4	26.0%	2.3%	8.0%	28.6%	12.5%	29.2%	-0.5%
	GPT-3.5	26.3%	3.3%	3.1%	7.6%	3.6%	4.4%	0.7%
	Claude 2	83.1%	6.0%	38.3%	15.1%	48.0%	14.4%	0.7%
	Beaver-7B	95.3%	60.7%	9.9%	12.1%	7.8%	7.6%	5.4%
	Alpaca-7B	97.7%	58.5%	5.8%	45.0%	22.6%	65.3%	10.0%
	Vicuna-7B	44.9%	58.5%	16.9%	15.8%	17.7%	27.1%	4.9%
	Vicuna-13B	53.9%	24.0%	19.4%	14.9%	17.1%	16.1%	7.6%
	Vicuna-33B	45.7%	39.3%	24.4%	52.4%	26.9%	32.6%	5.1%
	Llama2-7B-Chat	88.1%	69.5%	25.4%	7.2%	11.3%	25.9%	3.9%
	Llama2-13B-Chat	85.3%	53.4%	18.4%	12.3%	18.6%	27.6%	1.2%
	Llama2-70B-Chat	86.7%	47.9%	17.8%	5.5%	21.3%	7.2%	10.8%
7B	Average	66.6%	36.4%	17.0%	19.7%	18.8%	23.4%	4.5%
	GPT-4	27.7%	6.2%	18.6%	25.8%	16.3%	28.0%	-1.2%
	GPT-3.5	25.6%	6.8%	9.3%	9.3%	8.4%	7.0%	0.0%
	Claude 2	90.4%	10.4%	58.4%	30.3%	69.4%	42.1%	2.0%
	Beaver-7B	98.3%	83.5%	21.9%	12.0%	8.9%	6.0%	10.3%
	Alpaca-7B	99.4%	78.7%	34.9%	47.0%	38.2%	70.7%	11.8%
	Vicuna-7B	95.4%	73.6%	26.4%	15.9%	12.0%	29.3%	2.7%
	Vicuna-13B	94.0%	49.5%	37.6%	16.6%	21.9%	18.9%	2.7%
	Vicuna-33B	89.3%	58.5%	51.0%	55.9%	-1.0%	33.6%	3.2%
	Llama2-7B-Chat	95.6%	98.8%	19.9%	7.4%	-5.7%	22.1%	1.5%
	Llama2-13B-Chat	96.6%	70.8%	20.1%	10.3%	15.5%	28.6%	1.7%
13B	Llama2-70B-Chat	95.0%	70.1%	5.2%	2.4%	-6.6%	4.1%	9.1%
	Average	82.5%	55.2%	27.6%	21.2%	16.1%	26.4%	4.0%
	GPT-4	42.6%	9.7%	33.9%	25.1%	25.1%	20.1%	-0.2%
	GPT-3.5	43.7%	15.6%	15.1%	10.9%	7.6%	7.7%	0.5%
	Claude 2	90.6%	17.2%	50.0%	30.0%	45.9%	28.6%	0.5%
	Beaver-7B	98.1%	87.6%	14.2%	19.1%	8.0%	11.6%	13.0%
	Alpaca-7B	99.0%	82.9%	8.5%	53.4%	3.4%	75.9%	16.9%
	Vicuna-7B	96.3%	78.5%	19.1%	24.0%	19.5%	31.0%	6.6%
	Vicuna-13B	95.9%	58.7%	31.8%	26.7%	30.9%	18.9%	7.1%
	Vicuna-33B	90.0%	65.9%	33.3%	63.3%	7.3%	33.3%	6.1%
	Llama2-7B-Chat	96.0%	99.1%	13.5%	4.6%	12.6%	32.3%	4.2%
13B	Llama2-13B-Chat	95.4%	73.1%	16.7%	10.6%	30.7%	35.0%	1.0%
	Llama2-70B-Chat	94.6%	69.2%	10.6%	1.9%	6.3%	7.6%	10.3%
	Average	85.6%	59.8%	22.4%	24.5%	17.9%	27.4%	6.0%

Black-Box Prompt Optimization: Aligning Large Language Models without Model Training

Previous work has focused on aligning the output of large language models (LLMs). However, this paper explores the possibility of aligning the prompt to do the same thing.



BPO is Applicable for Various LLMs



The training objective is

$$\mathcal{L} = -\frac{1}{N} \sum_{t=1}^N \log P(x_t \mid X_{\text{user}}, x_{<t})$$

which is the same as SFT stage.

The training dataset is obtained from the method below:

1. Collect $(X, Y_{\text{good}}, Y_{\text{bad}})$, where Y_{good} stands for good response, then use this and LLM to generate a optimized prompts (see step 2).
2. Then use $(X_{\text{user}}, X_{\text{opt}})$ to train the seq-seq model (BPO) in a SFT way.
3. The seq-to-seq model is Llama-7B in the paper.

Result

Base LLM	Method		Vicuna Eval			Self-instruct Eval			Dolly Eval			BPO-test Eval			ΔWR
	A	B	A win	tie	B win	A win	tie	B win	A win	tie	B win	A win	tie	B win	
gpt-3.5-turbo	BPO ori.		60.0	8.7	31.3	50.4	12.3	37.3	55.0	16.0	29.0	51.0	18.0	31.0	+22.0
gpt-4	BPO ori.		41.3	23.7	35.0	39.7	22.6	37.7	51.0	26.0	23.0	39.0	26.0	35.0	+10.1
claude-instant-1.2	BPO ori.		66.3	5.0	28.7	50.0	9.1	40.9	45.0	14.5	40.5	45.0	10.5	44.5	+12.9
claude-2	BPO ori.		57.5	5.0	37.5	48.8	12.7	38.5	44.5	13.0	42.5	45.0	13.0	42.0	+8.8
text-bison	BPO ori.		65.0	10.0	25.0	47.0	21.9	31.1	42.0	30.5	27.5	50.5	10.5	39.0	+20.5

Figure 5: Result compared with original LLM

Result

Base LLM	Method		Vicuna Eval			Self-instruct Eval			Dolly Eval			BPO-test Eval			Δ WR
	A	B	A win	tie	B win	A win	tie	B win	A win	tie	B win	A win	tie	B win	
vicuna -7b-v1.3	PPO	ori.	47.5	10.0	42.5	49.6	10.3	40.1	46.0	13.9	38.5	42.0	19.5	36.0	+7.0
	BPO	PPO	61.3	6.2	32.5	49.6	11.9	38.5	49.0	12.5	41.5	47.5	13.0	39.5	+13.8
	BPO+PPO	ori.	55.0	7.5	37.5	50.0	10.3	39.7	52.5	9.0	38.5	54.5	10.0	35.5	+15.2
	BPO+PPO	PPO	56.3	11.2	32.5	44.4	20.7	34.9	43.0	29.0	28.0	44.0	23.0	33.0	+14.8
	DPO	ori.	58.8	6.2	35.0	53.6	11.5	34.9	50.0	19.0	31.0	51.0	18.0	31.0	+20.4
	BPO	DPO	53.8	3.7	42.5	40.1	8.3	51.6	45.0	10.0	45.0	45.0	11.0	44.0	+0.2
	BPO+DPO	ori.	65.0	5.0	30.0	60.3	10.7	29.0	54.0	17.0	29.0	56.0	13.0	31.0	+29.1
	BPO+DPO	DPO	63.8	2.5	33.7	49.6	9.9	40.5	46.0	14.0	40.0	45.0	16.0	39.0	+12.8
vicuna -13b-v1.3	PPO	ori.	53.8	3.7	42.5	49.2	11.1	39.7	49.0	14.5	36.5	42.0	17.5	40.5	+8.7
	BPO	PPO	52.5	3.7	43.7	44.4	6.4	49.2	50.0	9.0	41.0	53.5	11.5	35.0	+7.9
	BPO+PPO	ori.	55.0	7.5	37.5	49.6	9.9	40.5	54.0	11.0	35.0	55.5	11.5	33.0	+17.0
	BPO+PPO	PPO	55.0	5.0	40.0	49.6	5.6	44.8	49.5	9.5	41.0	55.0	11.0	34.0	+12.3
	DPO	ori.	50.0	3.7	46.3	55.6	6.3	38.1	58.5	6.5	35.0	58.0	11.5	30.5	+18.1
	BPO	DPO	53.8	2.5	43.7	44.0	8.4	47.6	45.0	5.0	50.0	43.0	16.0	41.0	+0.9
	BPO+DPO	ori.	71.3	2.5	26.2	61.1	7.2	31.7	58.0	9.0	33.0	62.0	8.0	30.0	+32.9
	BPO+DPO	DPO	60.0	2.5	37.5	48.8	9.1	42.1	48.0	8.5	43.5	50.0	11.0	39.0	+11.2

Figure 6: Win rates between PPO, DPO, and BPO-aligned LLMs