Alignment for LLMs

by yaoyhu

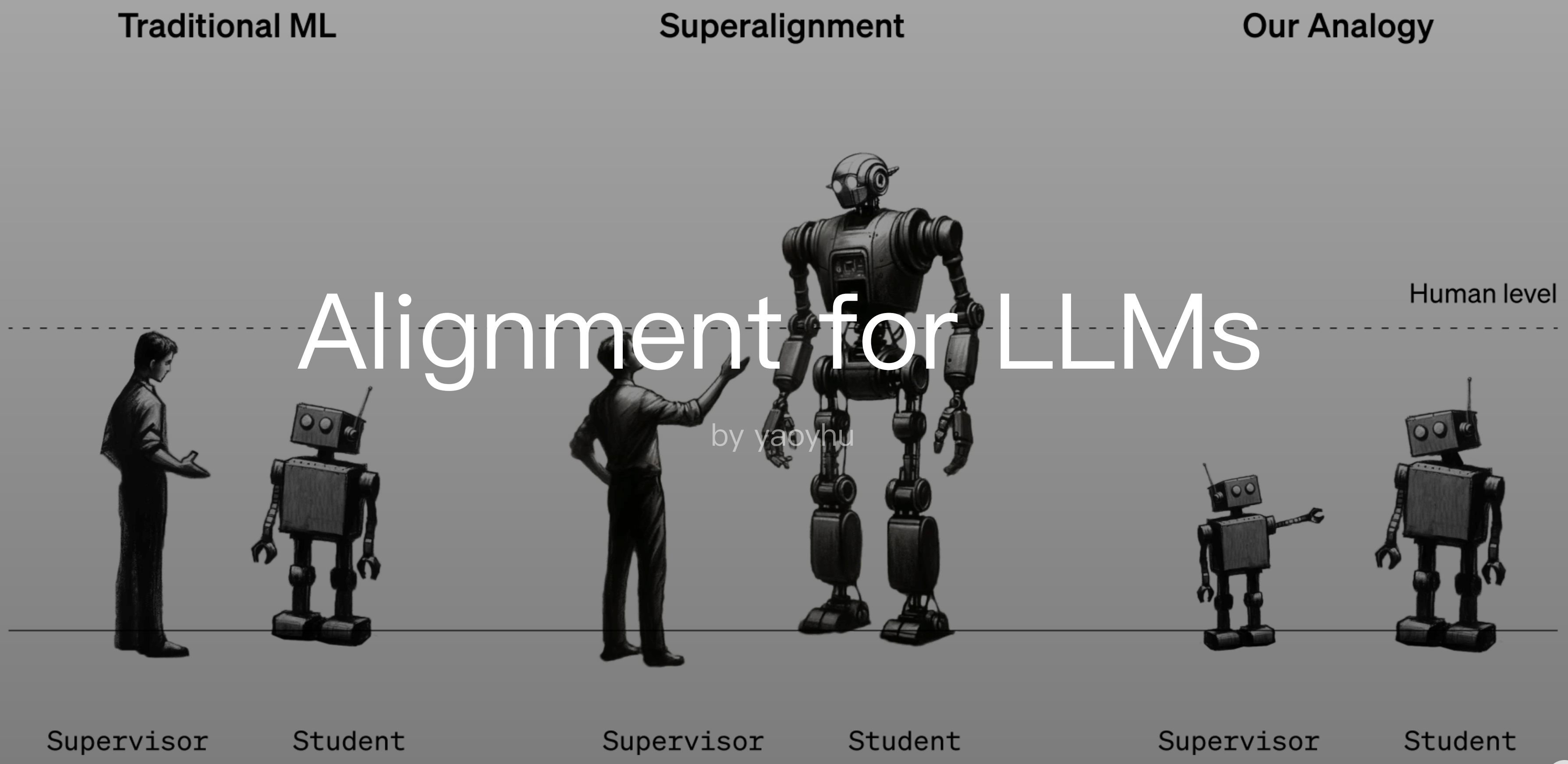


Table of contents

MedAligner

Med–Aligner empowers LLM medical applications for complex medical scenarios

1. Reliability
   - limited high–quality data
   - closed–source model rigidity
   - reasoning degradation during fine–tuning
2. Achievements
   - plug–and–play, even for closed–source models
   - without requiring full re–optimization
   - significant enhancements across all 3H dimensions—helpfulness, harmlessness, and honesty
3. No technical details (even wrong huggingface link)

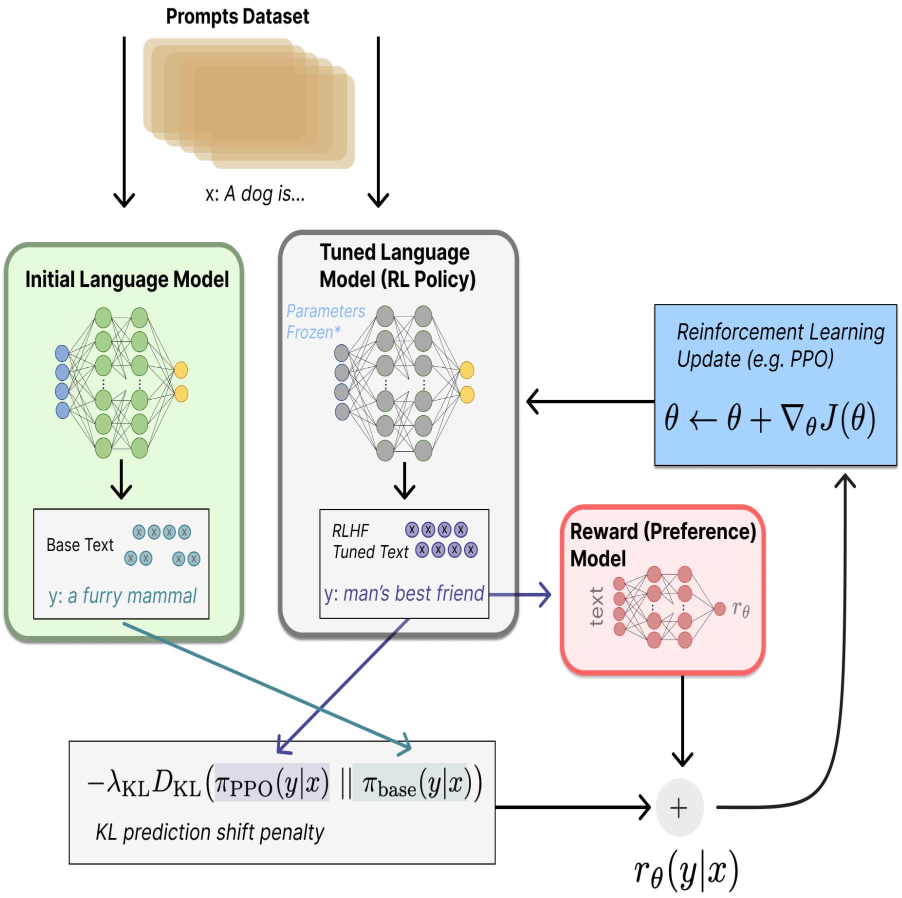
AI Alignment

How can we build AI systems that behave in line with human intentions and values?

1. Training process of LLMs:
   - **Pre-training**: Utilize large-scale text data to train a model for general capabilities through an autoregressive approach.
   - **Post-training**: Align the pre-trained model with specific tasks using instruction fine-tuning and **reinforcement learning with human feedback** (RLHF).
2. OpenAI demonstrated that RLHF enabled a smaller 1.3B parameter model to outperform a much larger 175B model.

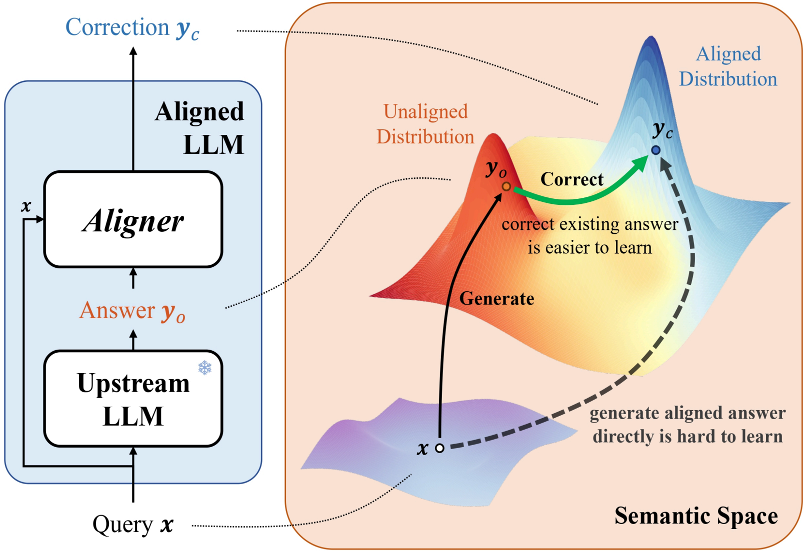
Reinforcement Learning with Human Feedback (RLHF)

1. Requires access to model parameters
2. Optimization redundancy: target model, reward model, critic model…
3. Full parameter tuning is challenging
4. Reward models have poor generalization
5. Alignment objectives are hard to define



Aligner: Efficient Alignment by Learning to Correct

1. Correction is easier than generation
2. A lightweight model to correct the target model's response
   - applicable across different base models
   - Completely bypassing RLHF, Aligner requires only a single line of code modification from SFT
   - For a 70B-parameter model, using Aligner saves 22.5 times the resources compared to RLHF.



The training process of Aligner

1. SFT: high-quality instruction dataset: $\{x^{(i)}, y^{(i)}; i = 1, \ldots, n\}$

$$\min_{\theta} \mathcal{L}(\theta; \mathcal{D}_{\text{sft}}) = -\mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{sft}}}[\log \pi_\theta(y|x)]$$

2. Aligner: $\{x^{(i)}, y_o^{(i)}, y_c^{(i)}; i = 1, \ldots, n\}$

- Copy: directly output the response from the upstream model
- Correction:residual correct the response from the upstream model.

$$\min_{\phi} \mathcal{L}_{\text{aligner}}(\phi; \mathcal{M}) = -\mathbb{E}_{\mathcal{M}}[\log \mu_\phi(y_c \mid y_0, x)]$$

Experiment Results

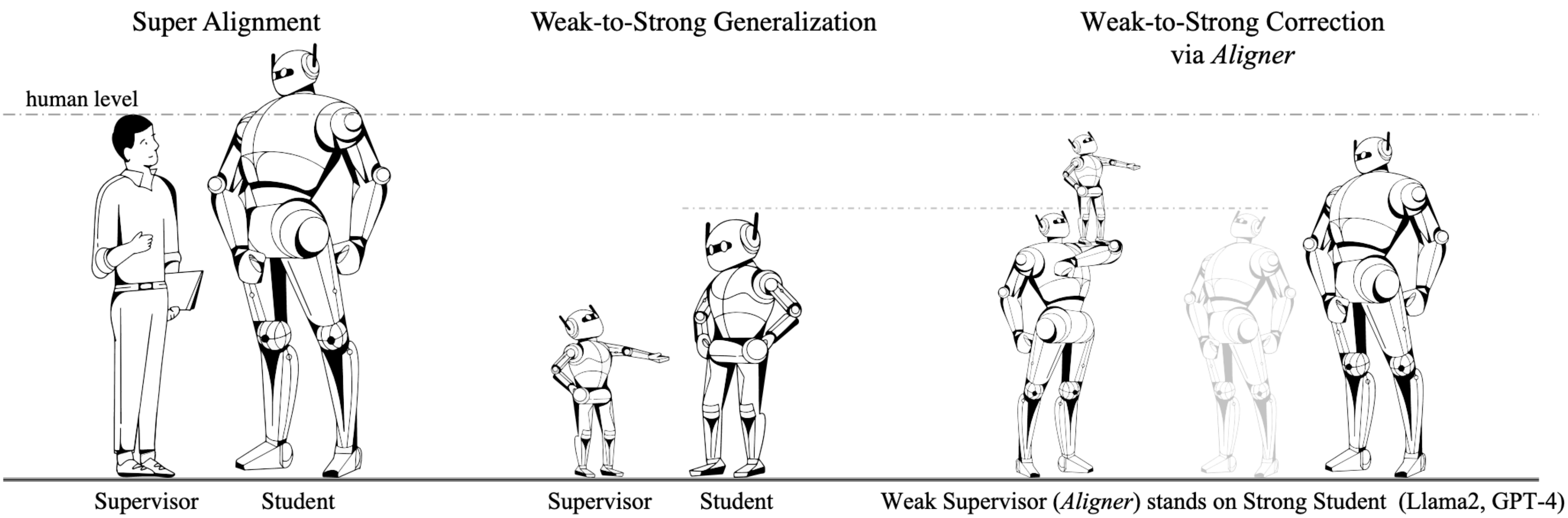
1. Improvements:
   - Improved Helpfulness
   - Enhanced Harmlessness
   - Reduced Hallucinations
2. Aligner exhibits a Scale Law trend as the model parameters increase
3. ~~bigger is better~~: e.g. the improvement from 2B to 13B is significant, but the gain from 13B to 70B is relatively limited.

Table 4: *Weak-to-Strong generalization* results demonstrate that *Aligner*-7B can achieve weak-to-strong generalization on 7B, 13B, and 70B upstream models with existing alignment methods using the labels given by the *Aligner*. This process entails enhancing the capabilities of a strong model by finetuning it with labels generated by a weak model.

	BeaverTails		HarmfulQA		Average	
Method†	Helpfulness	Harmlessness	Helpfulness	Harmlessness	Helpfulness	Harmlessness
Alpaca-7B **w/** *Aligner*-7B						
+SFT	+8.4%	**+53.5%**	+19.6%	**+73.9%**	+14.0%	+63.7%
+RLHF	-41.7%	**+51.4%**	-36.1%	**+73.9%**	-38.9%	+62.6%
+DPO	-48.2%	**+45.6%**	-54.4%	**+68.6%**	-51.3%	+57.1%
Alpaca2-13B **w/** *Aligner*-7B						
+SFT	**+34.7%**	**+49.4%**	+22.1%	**+69.7%**	+28.4%	+59.6%
+RLHF	**+46.0%**	+20.2%	-2.9%	**+67.6%**	+21.6%	+43.9%
+DPO	+1.3%	**+57.3%**	-20.4%	**+79.6%**	-9.6%	+68.4%
Alpaca2-70B **w/** *Aligner*-13B						
+SFT	+9.3%	**+46.9%**	+7.2%	**+76.3%**	+8.2%	+61.6%

† The weak-to-strong training dataset is composed of (q, a, a') triplets, with q representing queries from the *Aligner* training dataset-50K, a denoting answers generated by the Alpaca-7B model, and a' signifying the aligned answers produced by the *Aligner*-7B given (q, a). Unlike SFT, which solely utilizes a' as the ground-truth label, in RLHF and DPO training, a' is considered to be preferred over a.

Weak-to-Strong Generalization via Aligner



Building a Large Language Model

1. RAG: providing specific information for the model to draw from when answering questions
2. Fine-tuning: training the model on specific tasks
3. Alignment: ensuring the model's behavior aligns with human values and intentions