TechBeat Talk

权重视角下的大语言模型后训练过程
Rethinking LLM Post-train at Weight Level

Reporter: 吴太强 Taiqiang Wu

Supervisor: 黄毅 Prof. Ngai Wong

HKU EEE Ngai Lab

Nov. 2025

takiwu@connect.hku.hk

Agenda

- **背景介绍 Introduction**
  - LLM后训练 LLM Post-train
  - 为什么要关注权重 Why focus on weights
- **权重视角分析 Analysis at weight level**
  - 权重分布的相似 Similarity in weight values *(Paper: Shadow-FT)*
  - 权重有效秩的相似 Similarity in effective ranks *(Paper: Timber)*
- **相似性原因与应用 Why similar and application**
  - 相似的可能原因 Possible reasoning to explain weight similarity
  - 合并快慢思考模型实现高效思考 Application: model merge for efficient reasoning *(Paper: Revisit)*
- **未来方向 Future Work**

Agenda

- **背景介绍 Introduction**
  - LLM后训练 LLM Post-train
  - 为什么要关注权重 Why focus on weights
- **权重视角分析 Analysis at weight level**
  - 权重分布的相似 Similarity in weight values *(Paper: Shadow-FT)*
  - 权重有效秩的相似 Similarity in effective ranks *(Paper: Timber)*
- **相似性原因与应用 Why similar and application**
  - 相似的可能原因 Possible reasoning to explain weight similarity
  - 合并快慢思考模型实现高效思考 Application: model merge for efficient reasoning *(Paper: Revisit)*
- **未来方向 Future Work**

Introduction

LLM 后训练 Post-train

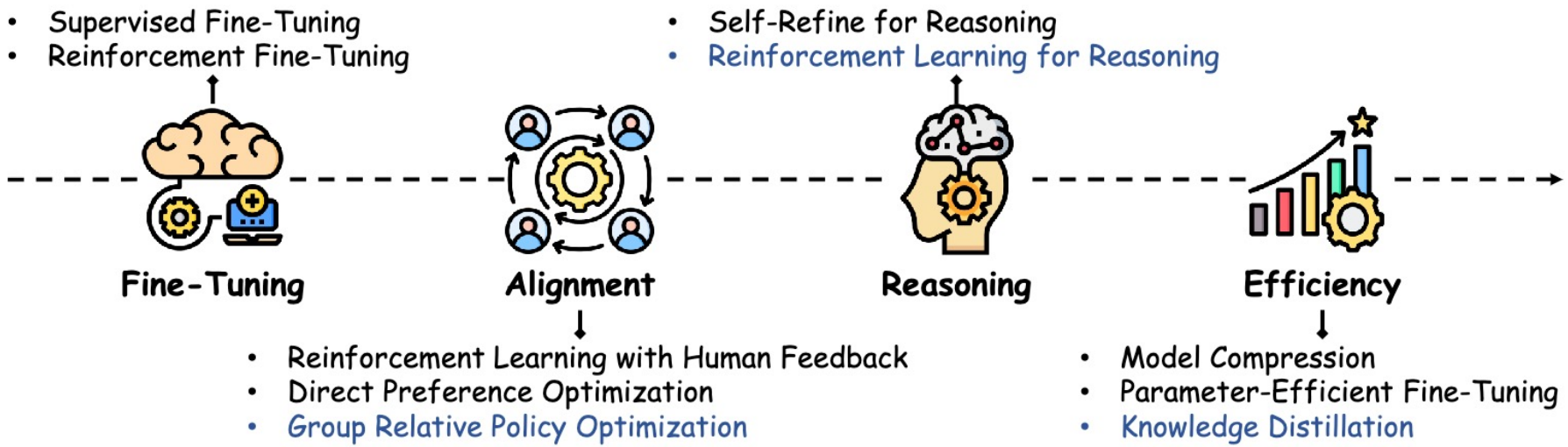
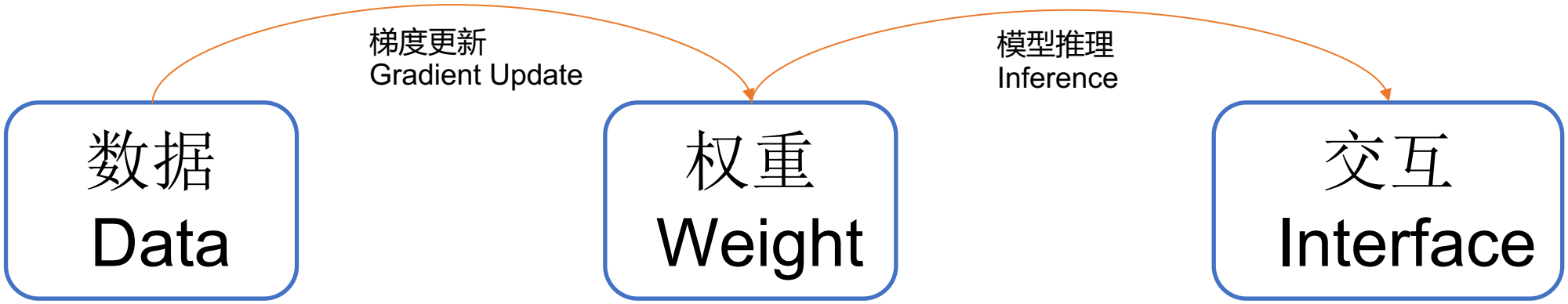


Figure 1: The evolution of post-training techniques for Large Language Models, delineating the progression from initial methodologies to advanced approaches, with emphasis on DeepSeek model contributions (highlighted in blue).

Base --> Chat/Instruct/Thinking

https://arxiv.org/pdf/2503.06072

为什么关注权重 Why study weight



权重是数据（梯度）的累积，是交互（特征）的源头

Weight is the accumulation of data (gradient), origin of interface (feature)

https://arxiv.org/pdf/2503.06072

Agenda

- 背景介绍 Introduction
  - LLM后训练 LLM Post-train
  - 为什么要关注权重 Why focus on weights
- **权重视角分析 Analysis at weight level**
  - 权重分布的相似 Similarity in weight values *(Paper: Shadow-FT)*
  - 权重有效秩的相似 Similarity in effective ranks *(Paper: Timber)*
- 相似性原因与应用 **Why similar and application**
  - 相似的可能原因 Possible reasoning to explain weight similarity
  - 合并快慢思考模型实现高效思考 Application: model merge for efficient reasoning *(Paper: Revisit)*
- 未来方向 **Future Work**

Analysis at weight level

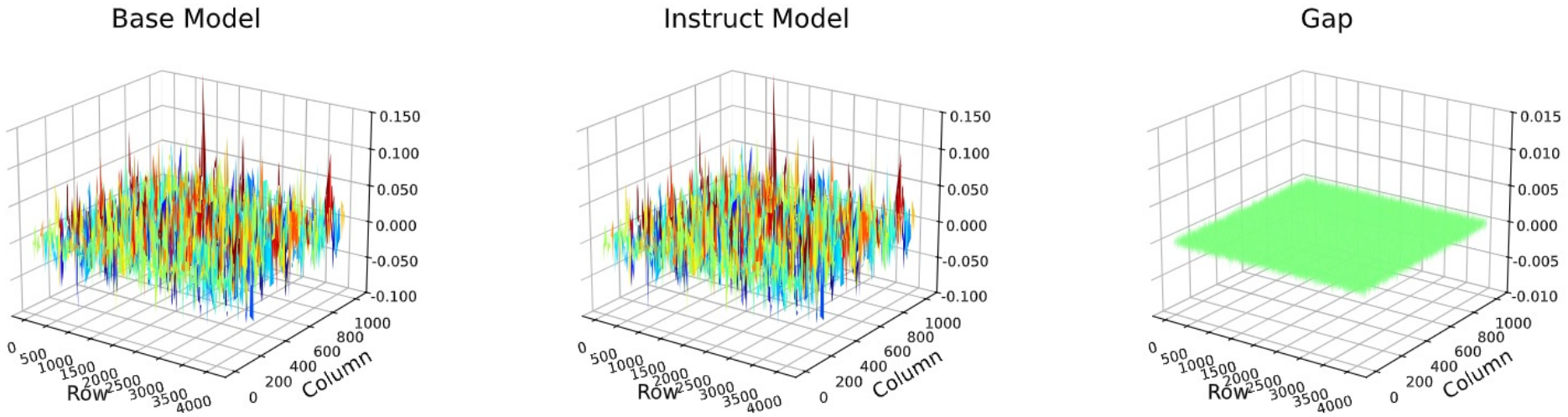


Figure 2: Weight distributions for Llama-3.1-8B. We visualize the same linear layer (layer.0.k_proj) for BASE model (left), INSTRUCT model (middle), and their gap (right). Though zoomed in 10x in the z-axis, the gap is negligible and the average σ value is 0.016.

线性层权重大小差异很小

High similarity between weight values

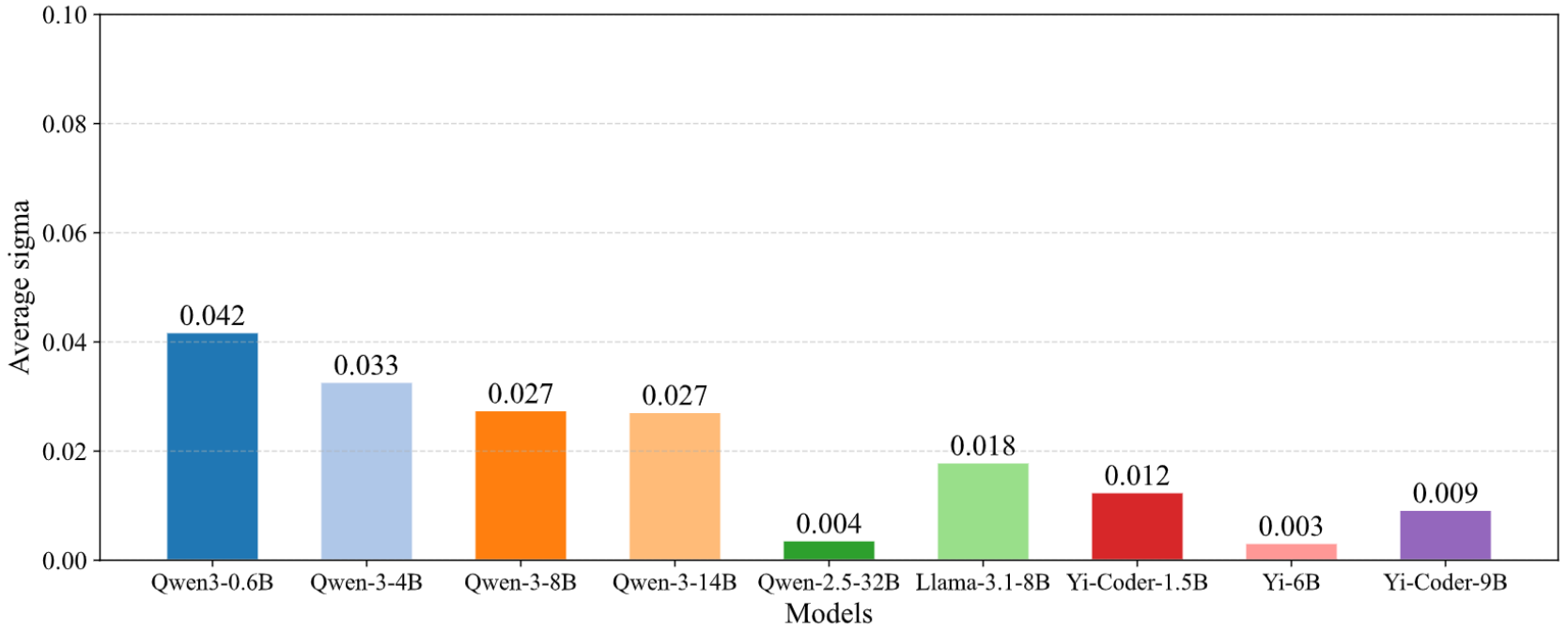
SHADOW-FT: TUNING INSTRUCT MODEL VIATRAINING ON PAIRED BASE MODEL

Analysis at weight level

$$\sigma = \frac{\sum |W_B - W_I|}{\sum |W_B| + \sum |W_I|},$$

定义 σ 定量衡量差异

Define σ for weight differences

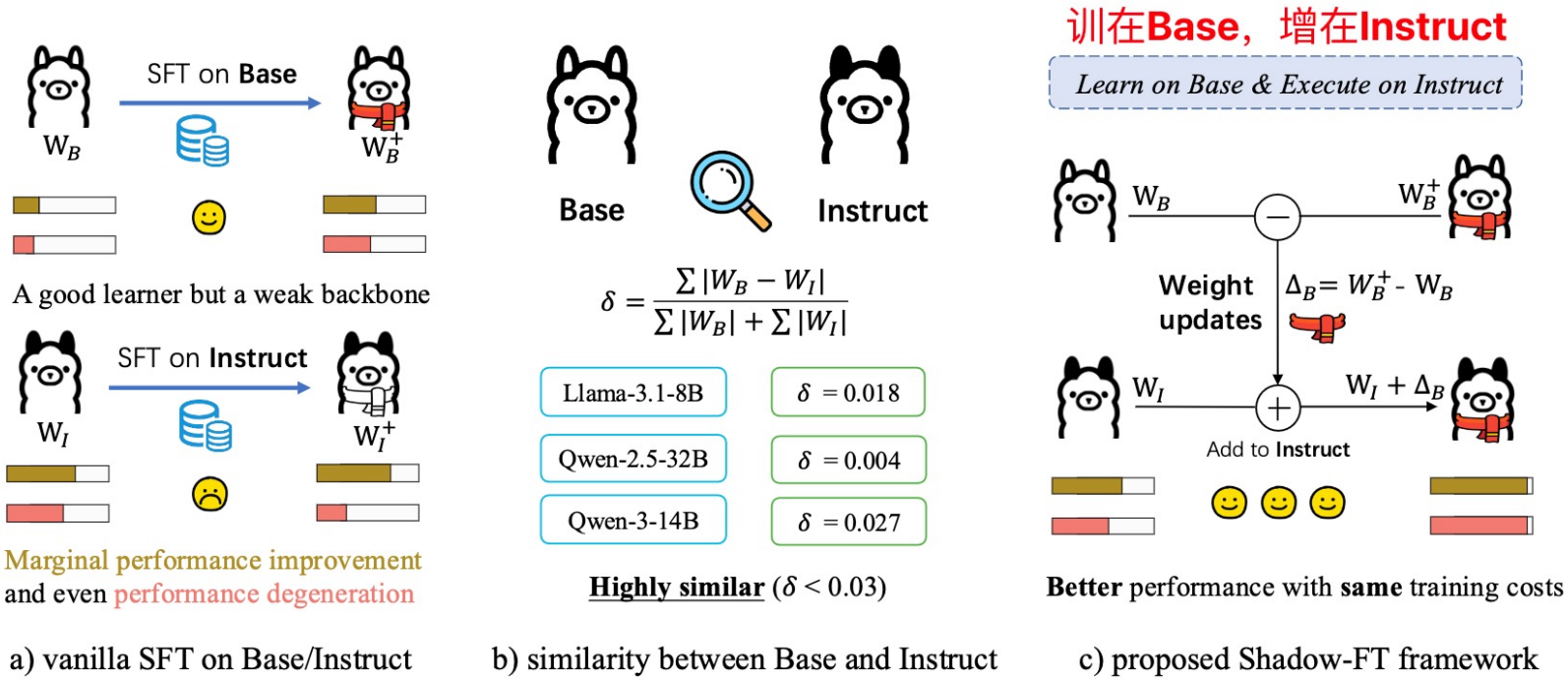


主流的 LLM 权重都高度相似

High similarity for mainstreaming LLMs

SHADOW-FT: TUNING INSTRUCT MODEL VIATRAINING ON PAIRED BASE MODEL

Analysis at weight level



用 Base 训练，将权重增量直接迁移给 Instruct 模型

Learn on Base model & Execute on Instruct model

SHADOW-FT: TUNING INSTRUCT MODEL VIATRAINING ON PAIRED BASE MODEL

Analysis at weight level

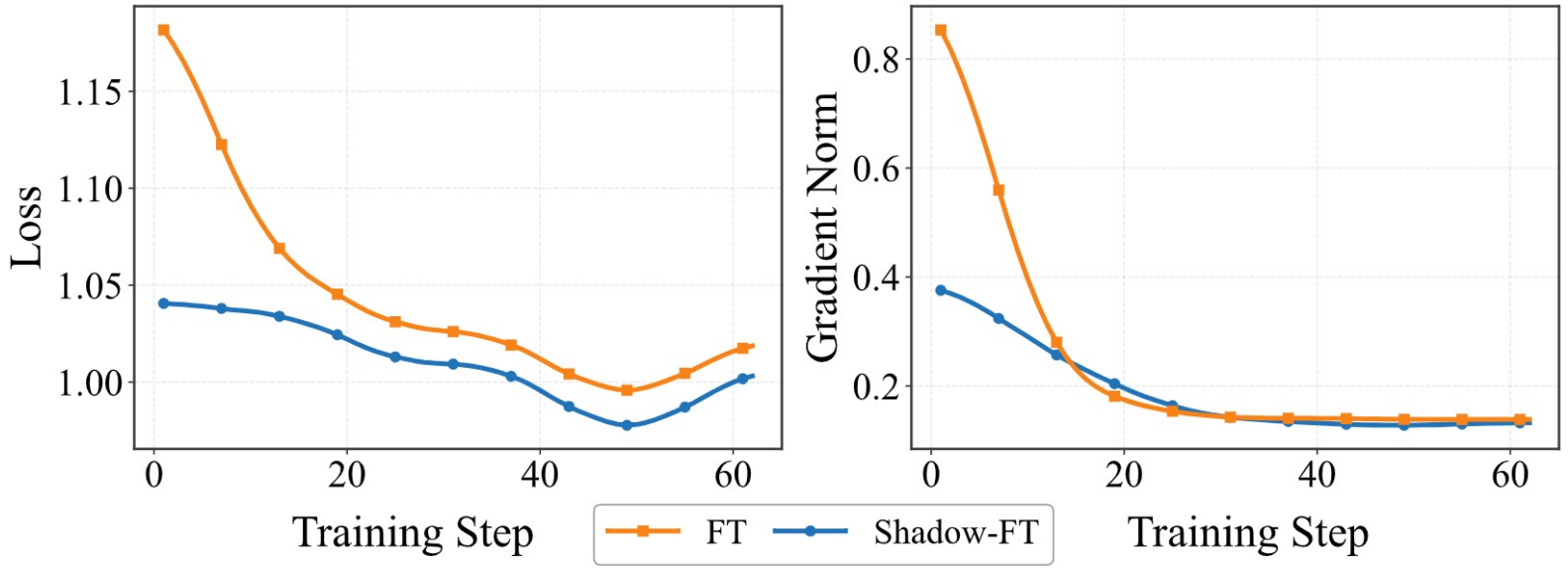


Figure 5: Optimization dynamics on loss and gradient when tuning Qwen3-8B via INSTRUCT (i.e., FT) and BASE (i.e., Shadow-FT).

Base 模型更适合学习新知识（梯度更小，loss 更低）

Base model is more suitable for learning new knowledge (lower grad & loss)

SHADOW-FT: TUNING INSTRUCT MODEL VIATRAINING ON PAIRED BASE MODEL

Analysis at weight level

权重值->权重特征（有效秩）

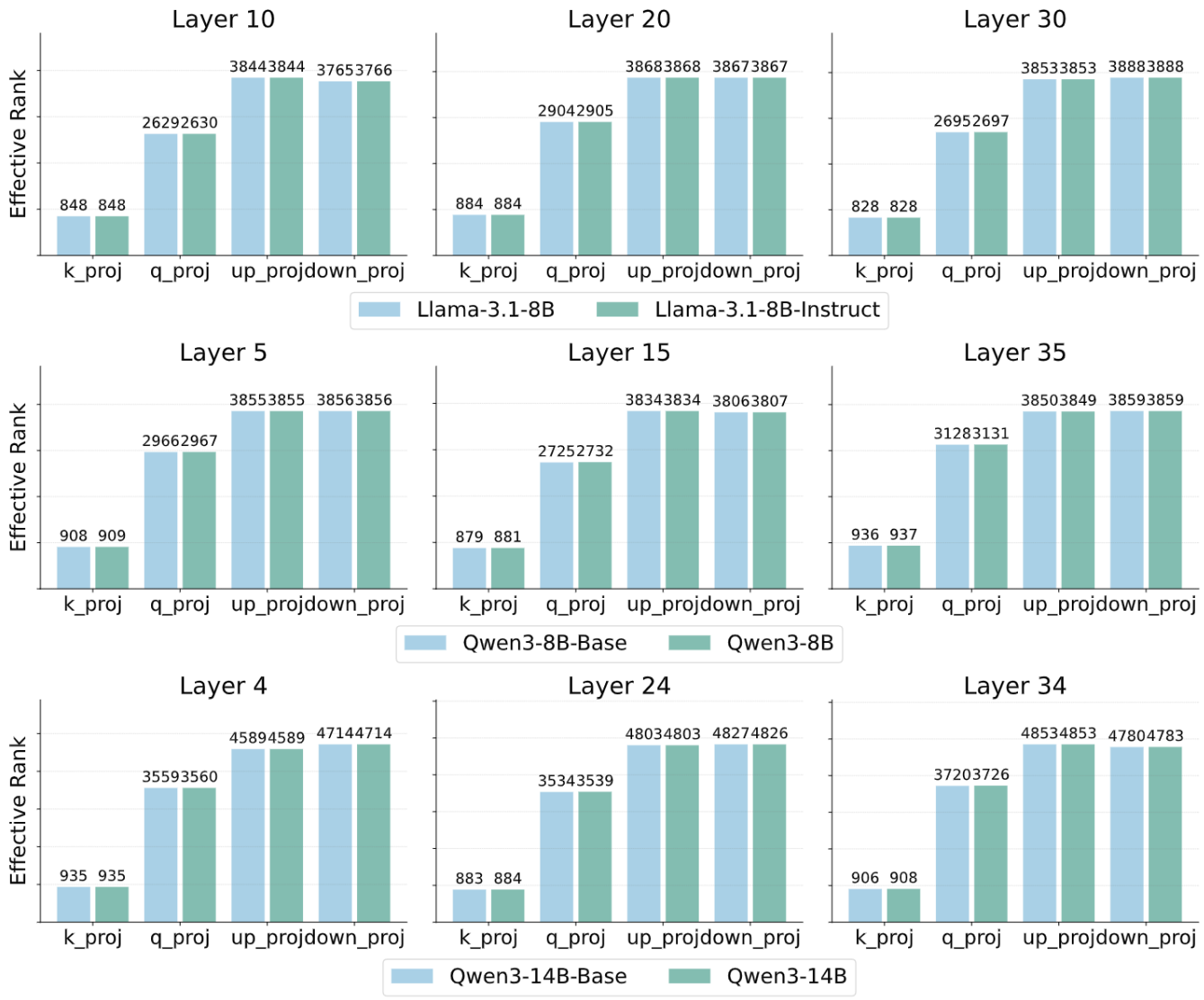
Values of weights -> Effective Rank of weights

$$\mathrm{eRank}(\mathbf{W}) = \exp\left(-\sum_{i=1}^{r} \frac{\sigma_i^{\gamma}}{\sum_{i=1}^{r} \sigma_i^{\gamma}} \log\left(\frac{\sigma_i^{\gamma}}{\sum_{i=1}^{r} \sigma_i^{\gamma}}\right)\right).$$

$$\text{singular values } \Sigma = \{\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_{r-1}, \sigma_r\}$$

TIMBER: TRAINING-FREE INSTRUCT MODEL REFIN-ING WITH BASE VIA EFFECTIVE RANK

Analysis at weight level

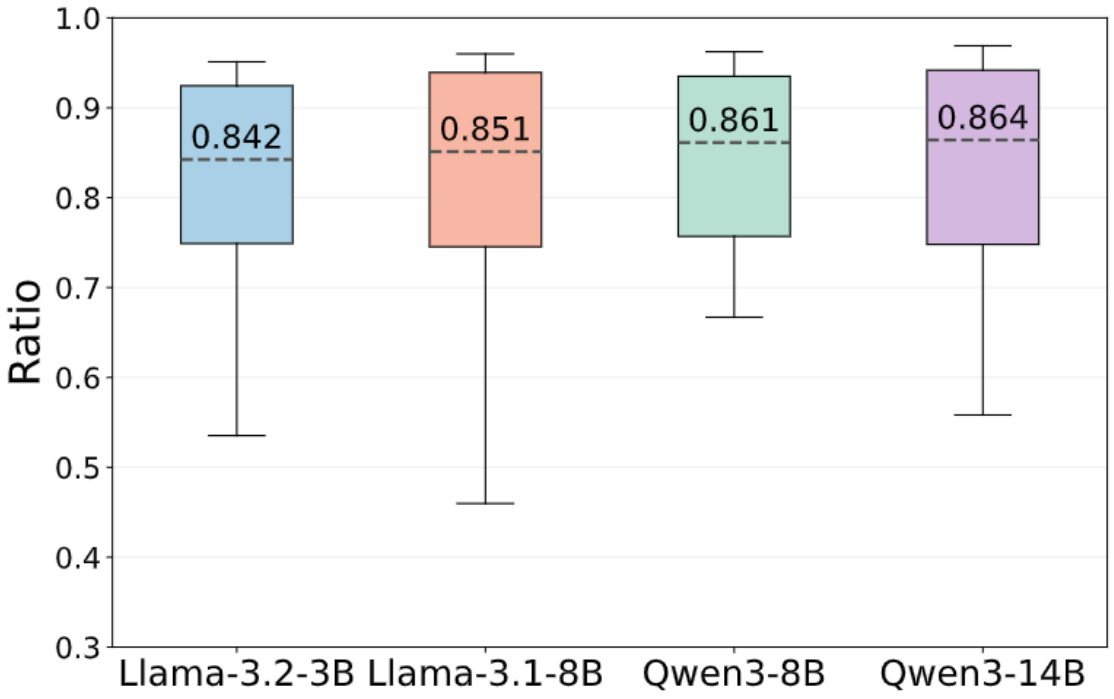


Base/Instruct 模型有效秩基本不变

Effective Ranks are almost the same for Base and Instruct counterparts

TIMBER: TRAINING-FREE INSTRUCT MODEL REFIN-ING WITH BASE VIA EFFECTIVE RANK

Analysis at weight level



有效秩是模型秩的 85% 附近

Effective ranks are around 85%

of weight ranks

Analysis at weight level

$$\text{eRank}(W_b) \approx \text{eRank}(W_i) \approx \text{eRank}(W_i - W_b)$$

$$W_b + W_\delta = W_i$$

$$\text{eRank}(W_b) + \text{eRank}(W_\delta) \geq \text{eRank}(W_i)$$

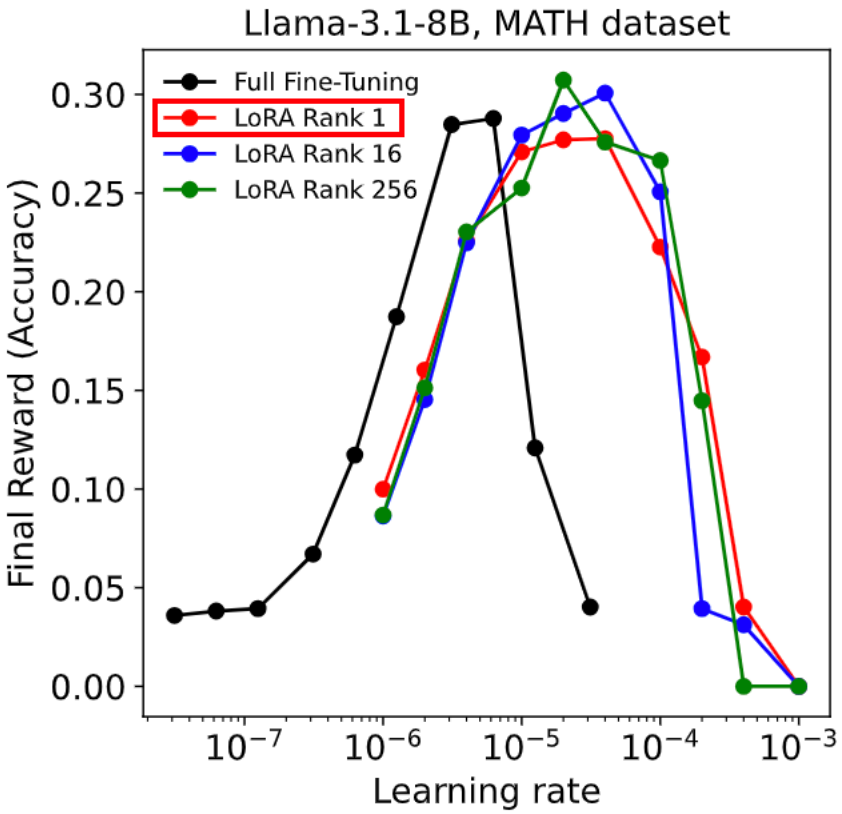
$$\rightarrow \text{eRank}(W_\delta) \geq \text{eRank}(W_i) - \text{eRank}(W_b) \approx 0$$

如果 W_δ 是 LoRA 结构，那么 eRank 约等于 Rank

If W_δ is LoRA, the $\text{eRank}(W_\delta) \approx$ LoRA Rank

-> 如果能用 LoRA 来进行后训练，那么秩的下限很小

(If we can use LoRA for post-train, then small rank works)



https://thinkingmachines.ai/blog/lora/

TIMBER: TRAINING-FREE INSTRUCT MODEL REFIN-ING WITH BASE VIA EFFECTIVE RANK

Agenda

- 背景介绍 Introduction
  - LLM后训练 LLM Post-train
  - 为什么要关注权重 Why focus on weights
- 权重视角分析 Analysis at weight level
  - 权重分布的相似 Similarity in weight values *(Paper: Shadow-FT)*
  - 权重有效秩的相似 Similarity in effective ranks *(Paper: Timber)*
- **相似性原因与应用 Why similar and application**
  - 相似的可能原因 Possible reasoning to explain weight similarity
  - 合并快慢思考模型实现高效思考 Application: model merge for efficient reasoning *(Paper: Revisit)*
- 未来方向 Future Work

Why similar and application

为什么后训练前后的权重如此相似？

Why weights before & after post-train are soooo similar

Reinforcement Learning from Human Feedback. Given the estimated reward function $r(\mathbf{x}, \mathbf{y})$, dictating the human preferences, RLHF fine-tunes policy $\pi_{\boldsymbol{\theta}}$ by optimizing the following objective:

$$\max_{\pi_{\boldsymbol{\theta}}} \mathbb{E}_{\mathbf{y} \sim \pi_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{x})} \left[r(\mathbf{x}, \mathbf{y}) \right] - \beta \mathbb{D}_{\mathrm{KL}} \left[\pi_{\boldsymbol{\theta}}(\mathbf{y} \mid \mathbf{x}) \| \pi_{\mathrm{ref}}(\mathbf{y} \mid \mathbf{x}) \right], \tag{2}$$

可能的原因#1 Possible reason #1

RLHF的KL项要求模型不要偏离太多 KL loss item as regularization

https://zhuanlan.zhihu.com/p/1922234702594539985

Why similar and application

为什么后训练前后的权重如此相似？

Why weights before & after post-train are soooo similar

可能的原因#2 Possible reason #2

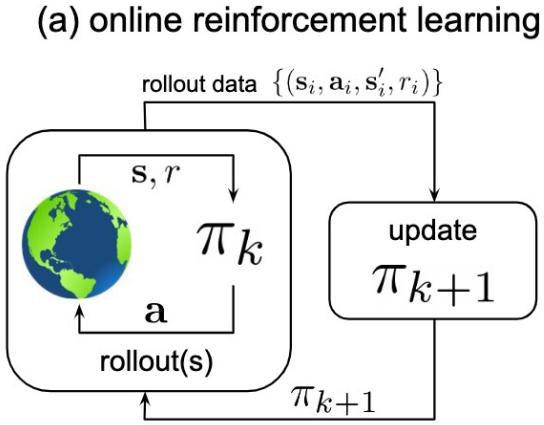
后训练数据相比预训练数据不多 Post-train data is less than pre-train

https://zhuanlan.zhihu.com/p/1922234702594539985

Why similar and application

为什么后训练前后的权重如此相似？

Why weights before & after post-train are soooo similar



$$\text{clip}\left(\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)}, 1-\epsilon, 1+\epsilon\right)$$

PPO-Clip

可能的原因#3 Possible reason #3

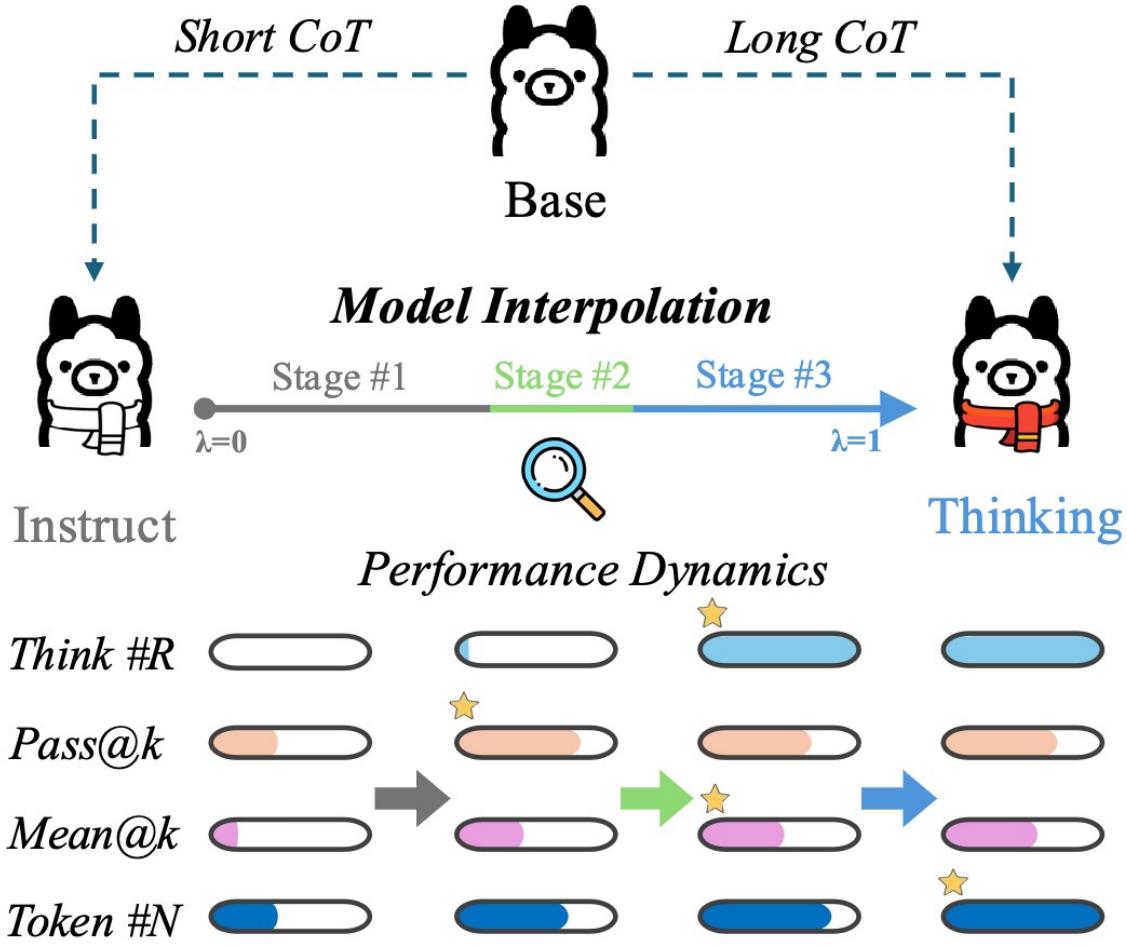
On-policy 策略本身更新就小 Marginal updates due to on-policy strategy

https://zhuanlan.zhihu.com/p/1922234702594539985

Why similar and application

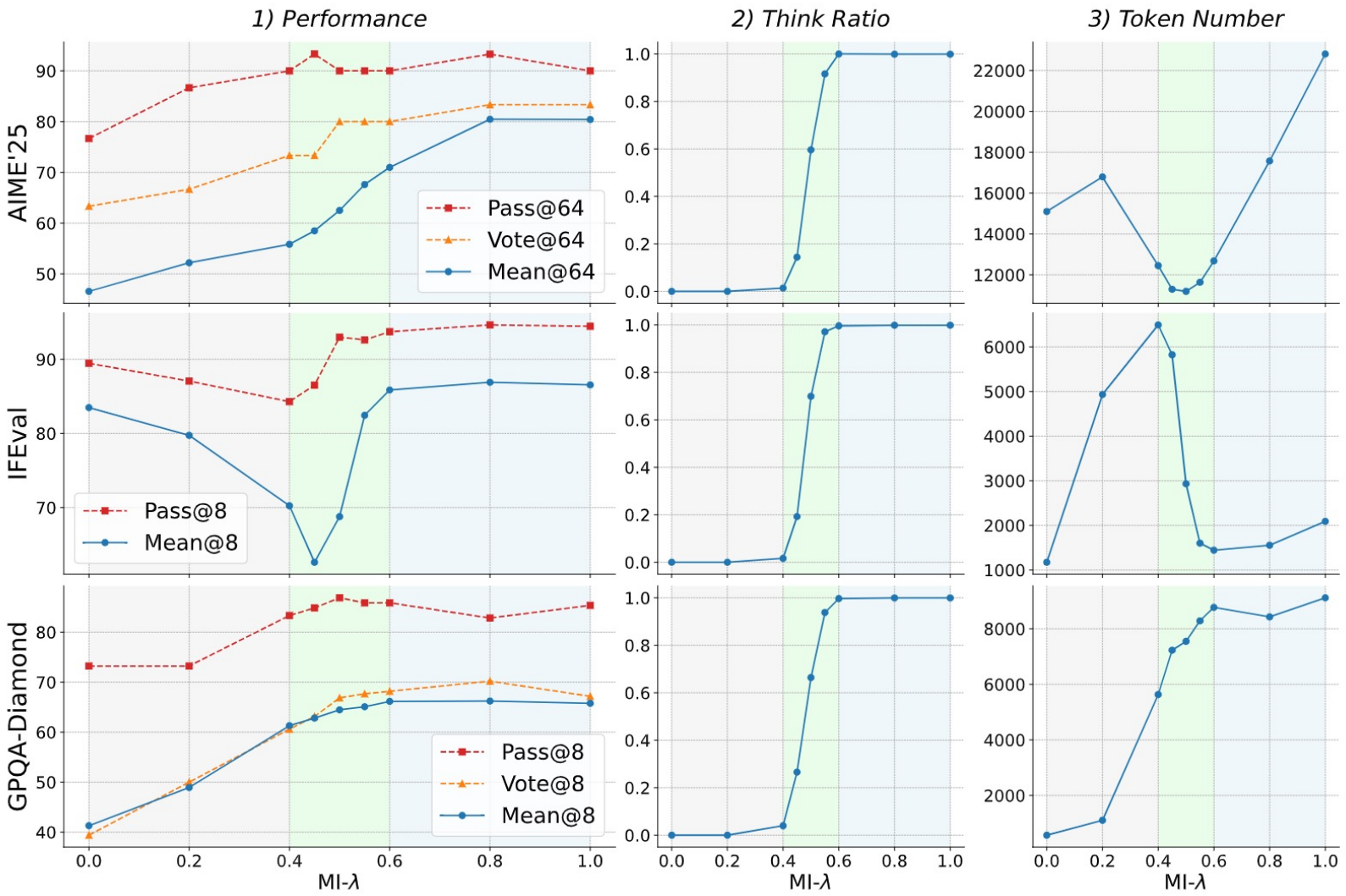
两个模型权重相似，则插值也相似

Interpolated weight would be similar if two weights are similar



Revisiting Model Interpolation for Efficient Reasoning

Why similar and application



插值 Instruct 和 Thinking 模型实现高效推理

Interpolating Instruct and Thinking model for Efficient Reasoning

Three-stage shifting

三阶段漂移

[Revisiting Model Interpolation for Efficient Reasoning]

Agenda

- 背景介绍 **Introduction**
  - LLM后训练 LLM Post-train
  - 为什么要关注权重 Why focus on weights
- 权重视角分析 **Analysis at weight level**
  - 权重分布的相似 Similarity in weight values *(Paper: Shadow-FT)*
  - 权重有效秩的相似 Similarity in effective ranks *(Paper: Timber)*
- 相似性原因与应用 **Why similar and application**
  - 相似的可能原因 Possible reasoning to explain weight similarity
  - 合并快慢思考模型实现高效思考 Application: model merge for efficient reasoning *(Paper: Revisit)*
- **未来方向 Future Work**

Future Work

- 基于相似性对后训练进行加速/直接免训练 How to speed up the post-train process or even training-free (post-train of other Base model/directed noise)

- 能否基于 Instruct 逆向工程获得 Base 模型 (Can we get Base version using Instruct version)　Qwen3-32B/GPT-OSS/Qwen3-Next

- 如何合并 基于同 Base 的 不同任务的后训练能力 (How to merge the post trained models on different tasks based on same Base model)