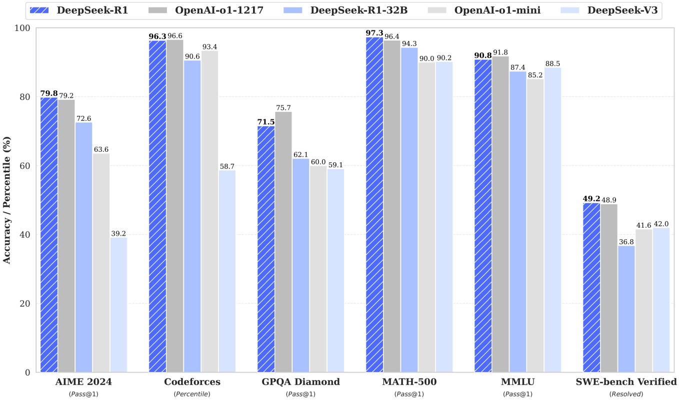
DeepSeek[1] Introduction

Lin Li

August 1, 2025

Overview

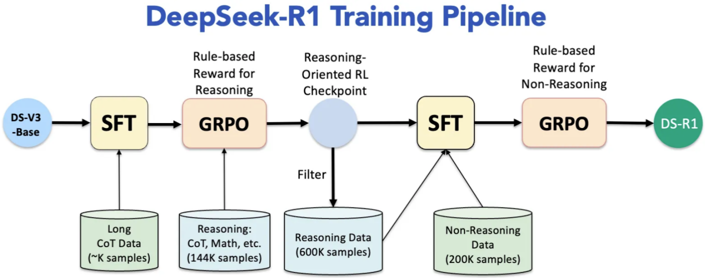
Benchmark Comparison



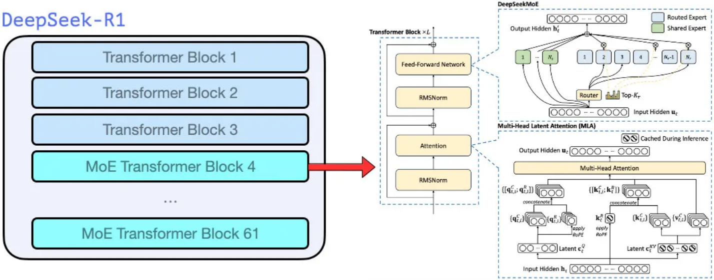
1/20th of the compute power achieved o1 similar performance

- Model architecture
  - MoE
  - Multi-headed latent attention
- Training Framework & optimizations
  - Mixed precision training with FP8
  - hardware optimizations for communication/computation overlap.

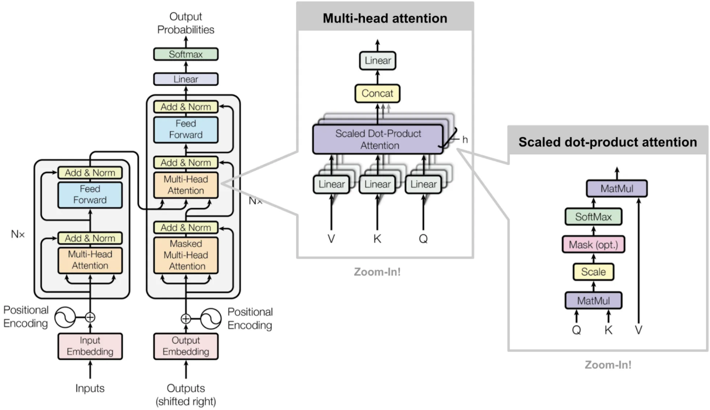
Training Pipeline



DeepSeek V3 workflow [6]

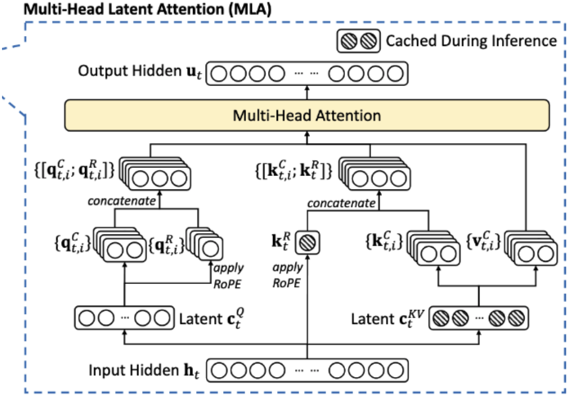


Default Attention



$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{1}$$

Multi-head Latent Attention (MLA)[2]

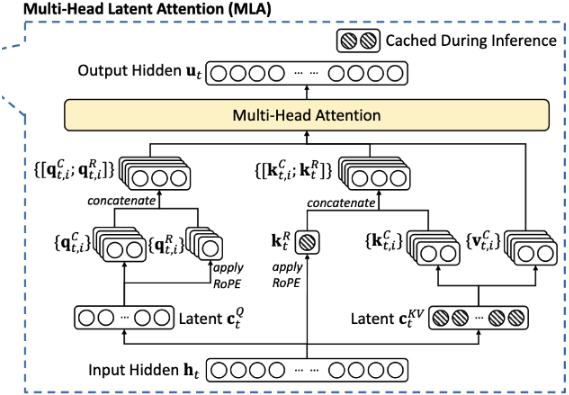


Key and Value in Attention:

$$c_t^{KV} = W^{DKV} \quad k_t^R = RoPE(W^{KR} h_t) \tag{2}$$

$$[k_{t,1}, k_{t,2}, \cdots, k_{t,n_h}] = k_t^C = W^{UK} c_t^{KV} \quad k_{t,i} = [k_{t,i}^C, k_t^R] \tag{3}$$

Multi-head Latent Attention (MLA)[2]

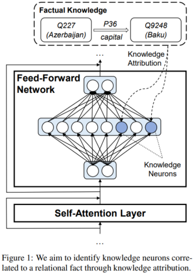


Query in Attention:

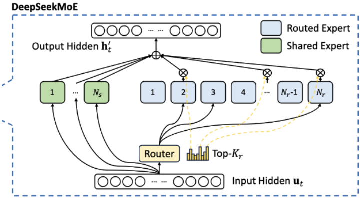
$$c_t^Q = W^{DQ} h_t \quad [q_{t,1}, q_{t,2}, \cdots, q_{t,n_h}] = q_t^C = W^{UQ} c_t^Q \tag{4}$$

$$q_t^R = RoPE(W^{QR} c_t^Q) \quad q_{t,i} = [q_{t,i}^C, q_t^R] \tag{5}$$

Mixture-of-Experts (MoE)[2]



(a) Standard FFN



(b) MoE

$$h_t' = u_t + \sum_{i=1}^{N_s} FFN_i^{(s)}(u_t) + \sum_{i=1}^{N_r} g_{i,t} FFN_i^{(r)}(u_t) \tag{6}$$

$$g_{i,t} = \frac{g_{i,t}'}{\sum_{j=1}^{N_r} g_{j,t}'}, g_{i,t}' = s_{i,t}, s_{i,t} \in TopK(s_{j,k}|1 \leq j \leq N_r) \; else \; 0 \tag{7}$$

Multi-Token Prediction(MTP)

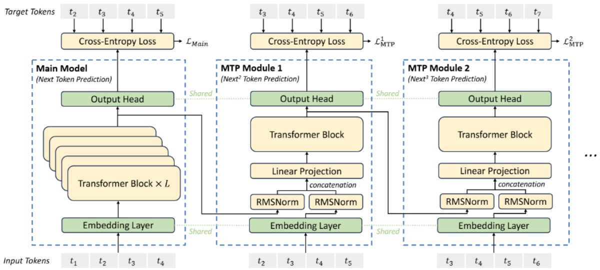


Figure 3 | Illustration of our Multi-Token Prediction (MTP) implementation. We keep the complete causal chain for the prediction of each token at each depth.

- Better Performance in Long-Form Text Generation
- Increased Efficiency & Speed

Supervised Fine Tuning(SFT)[3]

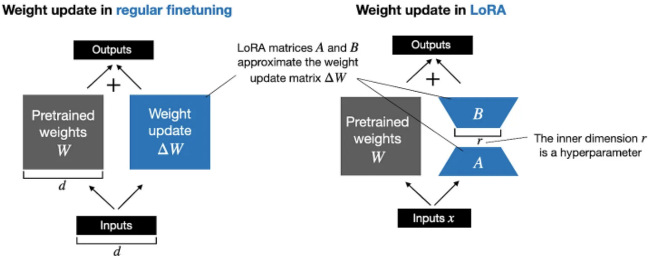
What are the benefits of supervised fine-tuning?

- Task-specific patterns and nuances
- Improved performance
- Data efficiency
- Resource efficiency
- Customization

What are some common supervised fine-tuning techniques?

- LoRA (Low-Rank Adaptation)
- QLoRA (Quantized LoRA)
- Few-shot learning

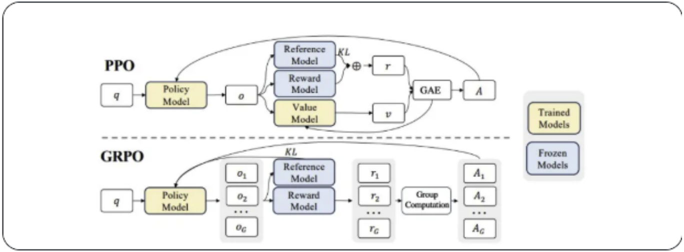
LoRa[5]



$$W_{update} = W + \Delta W \Longrightarrow W_{update} = W + AB \tag{8}$$

We fixed W and replace ΔW with low rank matrix A, B, if W is 1,000,000 dimension, we can use A is a 1000×2 matrix, and B is a 2×1000 matrix.

GRPO vs Proximal Policy Optimization(PPO)[4]



Advantage of GRPO vs PPO

- Unsupervised learning(without labeling)
- Avoid Value model (save training cost)

PPO loss:

$$\max_{\theta} E_{q \quad P(Q)}[\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}A_i] - \beta E_{q \quad P(Q)}[KL(\pi_\theta(o_i|q), \pi_{\theta_{old}}(o_i|q))] \tag{9}$$

GRPO loss

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{old}}(O|q)]$$
$$\frac{1}{G}\sum_{i=1}^{G}\left(\min\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}A_i, \text{clip}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}, 1-\varepsilon, 1+\varepsilon\right)A_i\right) - \beta\mathbb{D}_{KL}\left(\pi_\theta||\pi_{ref}\right)\right), \tag{1}$$

$$\mathbb{D}_{KL}\left(\pi_\theta||\pi_{ref}\right) = \frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - \log\frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - 1, \tag{2}$$

where ε and β are hyper-parameters, and A_i is the advantage, computed using a group of rewards $\{r_1, r_2, \ldots, r_G\}$ corresponding to the outputs within each group:

$$A_i = \frac{r_i - \text{mean}(\{r_1, r_2, \cdots, r_G\})}{\text{std}(\{r_1, r_2, \cdots, r_G\})}. \tag{3}$$

π_θ: new policy and $\pi_{\theta_{old}}$: old policy, $\pi_{\theta_{ref}}$: reference policy(constant).
q is query(input of model), o_i is i_{th} output.
r_i: reward for output o_i, which includes accuracy reward and format reward.

DeepSeek-R1-Zero Training Loss

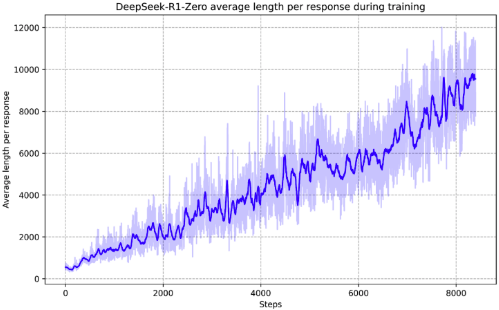


Figure 3 | The average response length of DeepSeek-R1-Zero on the training set during the RL process. DeepSeek-R1-Zero naturally learns to solve reasoning tasks with more thinking time.

Why a Second Round of RL Training

- The first RL stage primarily concentrated on accuracy and logical reasoning
- The second round of RL training was essential for refining the model's overall performance and ensuring alignment with human preferences

What Can we do with DeepSeek

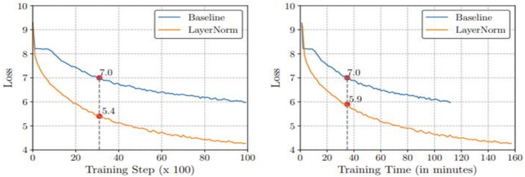
- Automating 510(k) Submission Preparation
- Automated Financial Analysis
- Automated Report Generation
- ...

Why we need to do it locally

- Data Security Compliance
- Cost Savings in the Long Run
- Faster Processing Efficiency
- Full Control Customization

Question

RMS[7] VS LayerNorm



(a) Training loss vs. training steps. (b) Training loss vs. training time.

LayerNorm:

$$\bar{a}_i = \frac{a_i - \mu}{\sigma} g_i \quad \mu = \frac{1}{n}\sum_{i=1}^{n} a_i \quad \sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(a_i - \mu)^2} \tag{10}$$

RMS:

$$\bar{a}_i = \frac{a_i}{RMS(a)} g_i \quad RMS(a) = \sqrt{\frac{1}{n}\sum_{i=1}^{n} a_i^2} \tag{11}$$

[1] DeepSeek-AI et al. *DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning*. 2025. arXiv: 2501.12948 [cs.CL]. URL: https://arxiv.org/abs/2501.12948.

[2] DeepSeek-AI et al. *DeepSeek-V3 Technical Report*. 2024. arXiv: 2412.19437 [cs.CL]. URL: https://arxiv.org/abs/2412.19437.

[3] Stephen M. Walker II. *Supervised fine-tuning (SFT)*. 2025. URL: https://klu.ai/glossary/supervised-fine-tuning.

[4] John Schulman et al. *Proximal Policy Optimization Algorithms*. 2017. arXiv: 1707.06347 [cs.LG]. URL: https://arxiv.org/abs/1707.06347.

[5] PhD Sebastian Raschka. *https://magazine.sebastianraschka.com/p/lora-and-dora-from-scratch*. 2024. URL: https://magazine.sebastianras com/p/lora-and-dora-from-scratch.

[6] Shakti Wadekar. *DeepSeek-R1: Model Architecture*. 2025 Feb. URL: https://shaktiwadekar.medium.com/deepseek-r1-model-architecture-853fefac7050.

[7] Biao Zhang and Rico Sennrich. "Root Mean Square Layer Normalization". In: *Advances in Neural Information Processing Systems 32*. Vancouver, Canada, 2019. URL: https://openreview.net/references/pdf?id=S1qBAf6rr.