

DeepSeek





Gliederung

- Architecture - Hannes & Johanna
 - DeepSeek MoE - Johanna
 - Multi-Head Latent Attention - Hannes
 - Multi-token Prediction - Hannes
- Infrastructure - Maria
 - Mixed Precision Framework (FP8)
 - Leistungsbeschränkungen
- Pre-Training - Emmanuel
- Benchmarks - Johanna
 - Post-Training - Johanna
- Conclusion



Folien



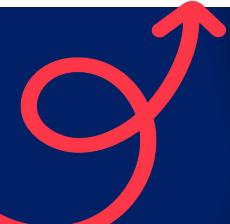


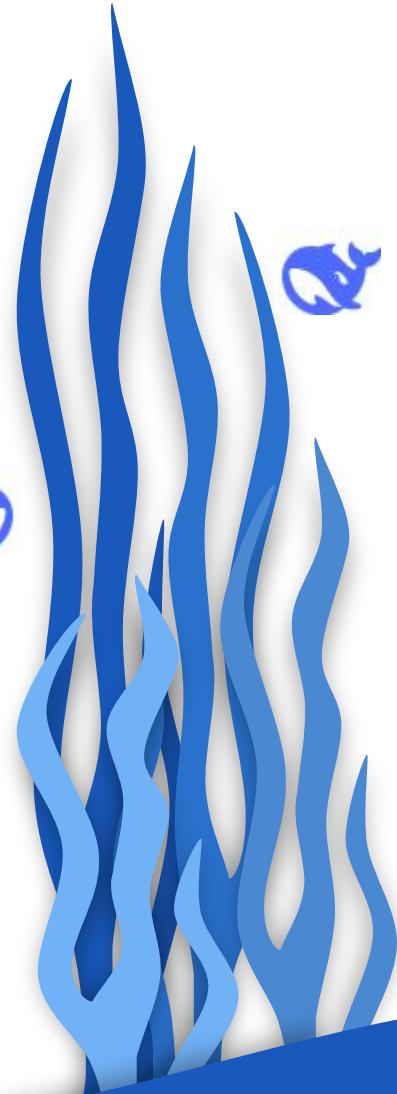
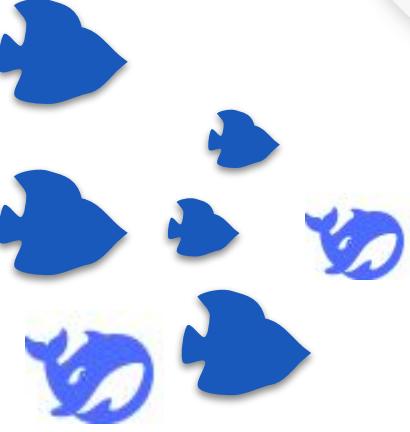
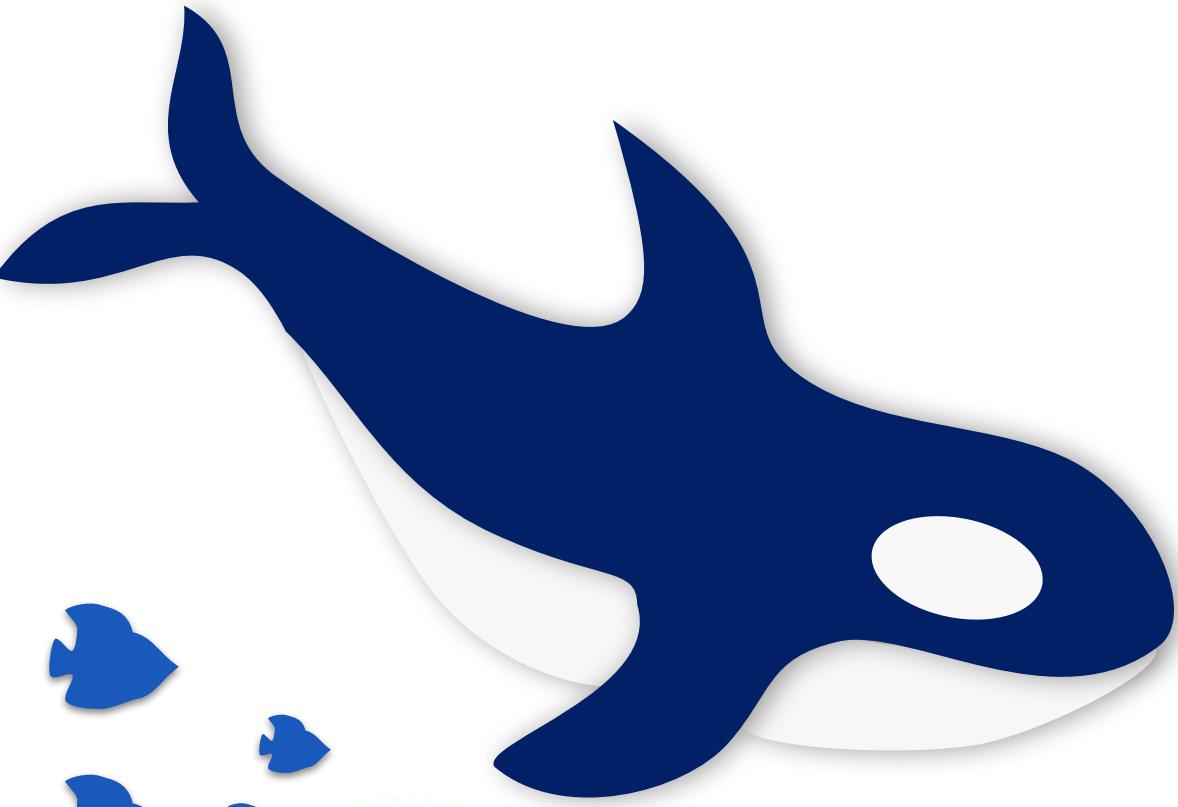
Gliederung

- 🙏 DeepSeek R1 Zero
 - Reinforcement Learning Algorithm - Maria/Johanna
 - Reward Modeling - Maria
 - Training Template - Maria
 - Evaluation - Maria
- 🤝 DeepSeek R1
 - Cold Start - Emmanuel
 - Distillation - Emmanuel



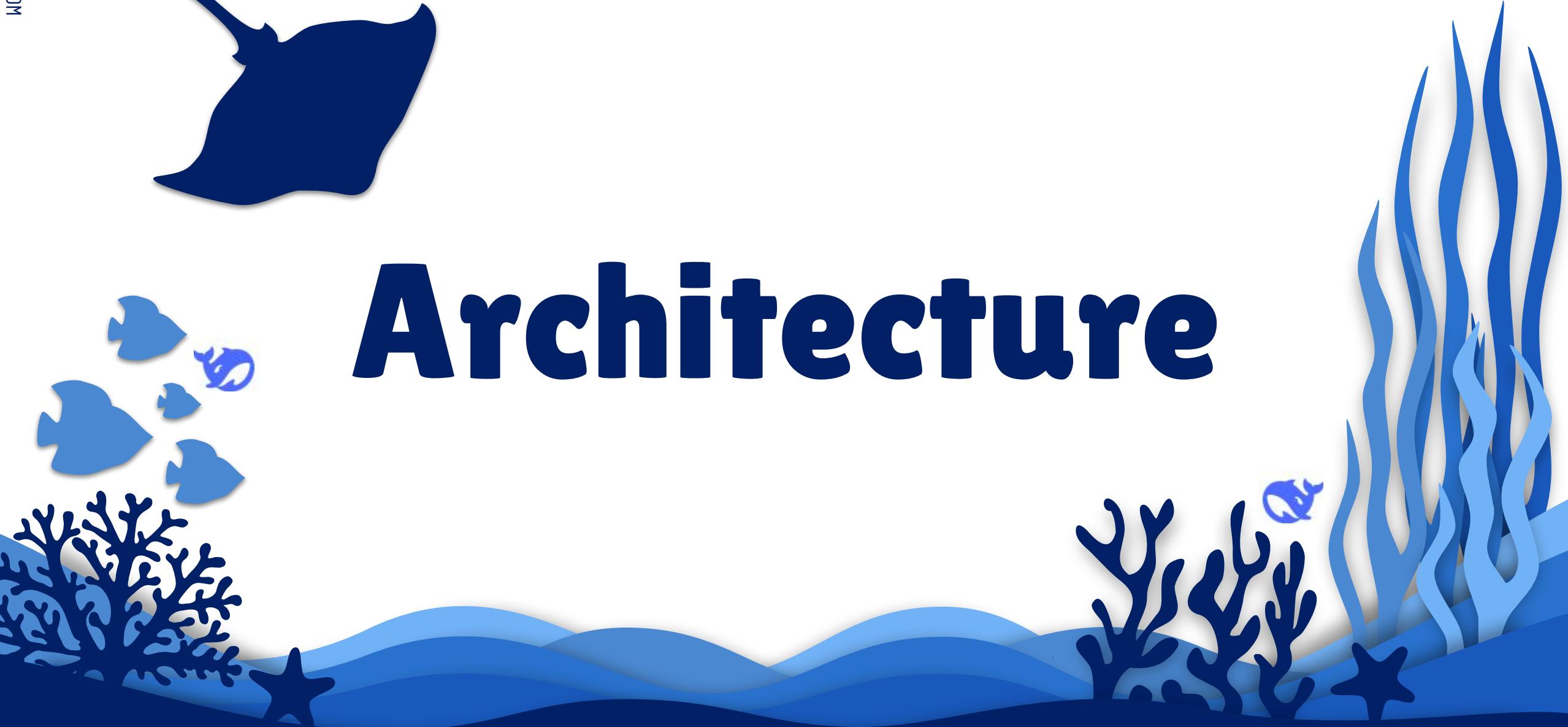
Folien





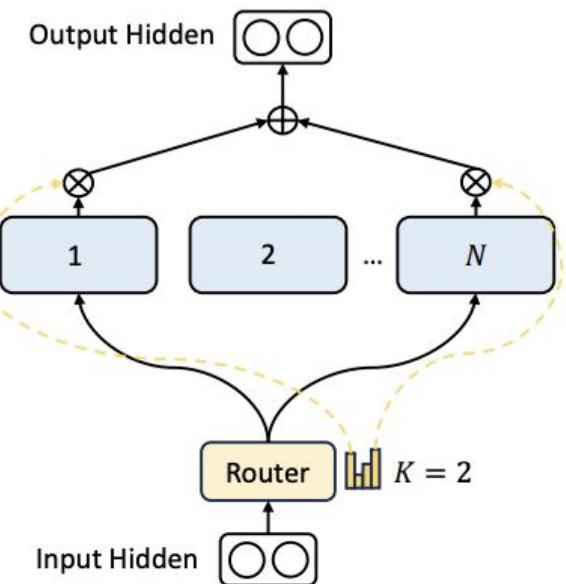
Previously on DeepSeek...

Architecture

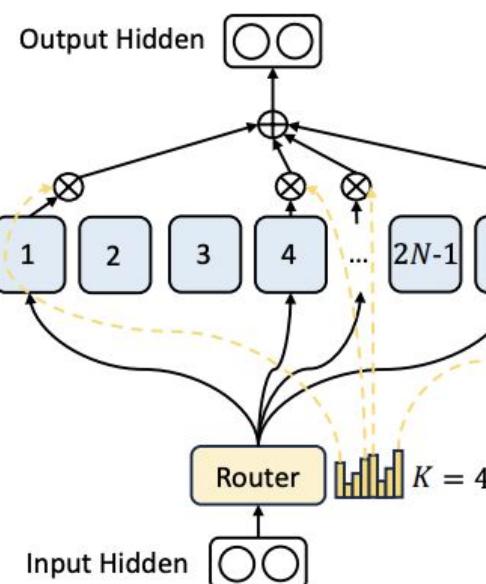


Mixture of Experts (MoE)

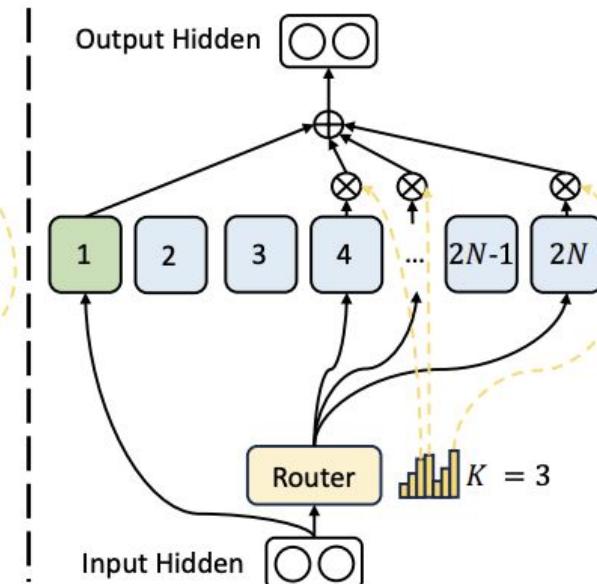




(a) Conventional Top-2 Routing



(b) + Fine-grained Expert Segmentation

(c) + Shared Expert Isolation
(DeepSeekMoE)

Auxiliary–Loss–Free Load Balancing

$$\mathbf{h}'_t = \mathbf{u}_t + \sum_{i=1}^{N_s} \text{FFN}_i^{(s)}(\mathbf{u}_t) + \sum_{i=1}^{N_r} g_{i,t} \text{FFN}_i^{(r)}(\mathbf{u}_t),$$

$$g_{i,t} = \frac{g'_{i,t}}{\sum_{j=1}^{N_r} g'_{j,t}},$$

$$g'_{i,t} = \begin{cases} s_{i,t}, & s_{i,t} \in \text{Topk}(\{s_{j,t} | 1 \leq j \leq N_r\}, K_r), \\ 0, & \text{otherwise,} \end{cases}$$

$$s_{i,t} = \text{Sigmoid}(\mathbf{u}_t^T \mathbf{e}_i),$$

$$\mathcal{L}_{\text{DevBal}} = \alpha_2 \sum_{i=1}^D f'_i P'_i,$$

$$f'_i = \frac{1}{|\mathcal{E}_i|} \sum_{j \in \mathcal{E}_i} f_j,$$

$$P'_i = \sum_{j \in \mathcal{E}_i} P_j,$$

$$\mathcal{L}_{\text{ExpBal}} = \alpha_1 \sum_{i=1}^{N_r} f_i P_i,$$

$$f_i = \frac{N_r}{K_r T} \sum_{t=1}^T \mathbb{1}(\text{Token } t \text{ selects Expert } i),$$

$$P_i = \frac{1}{T} \sum_{t=1}^T s_{i,t},$$

$$\mathcal{L}_{\text{CommBal}} = \alpha_3 \sum_{i=1}^D f''_i P''_i,$$

$$f''_i = \frac{D}{MT} \sum_{t=1}^T \mathbb{1}(\text{Token } t \text{ is sent to Device } i),$$

$$P''_i = \sum_{j \in \mathcal{E}_i} P_j,$$

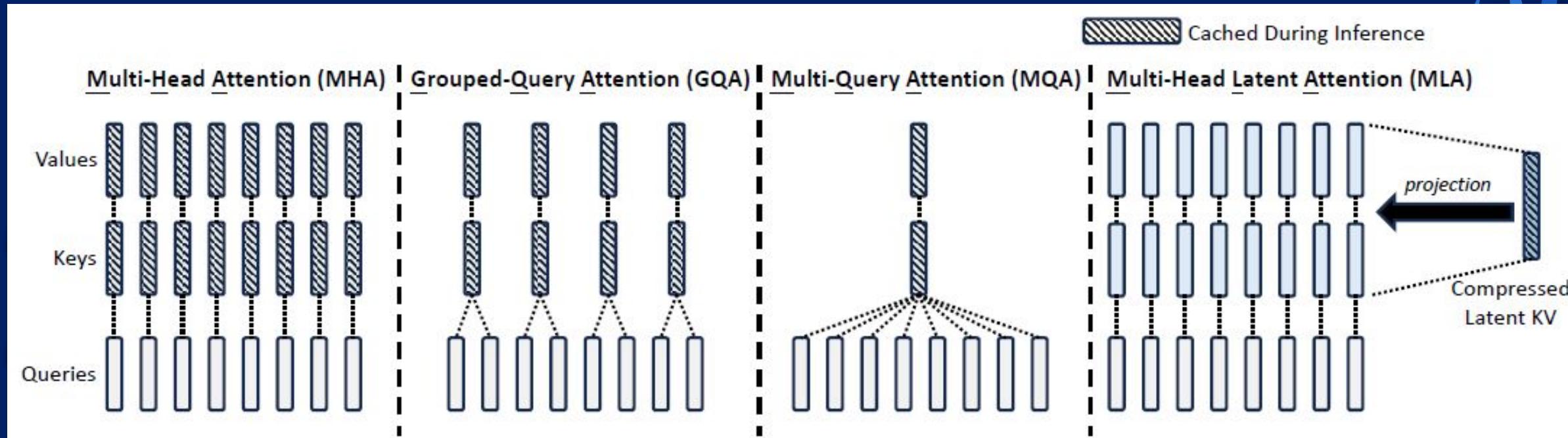


$$g'_{i,t} = \begin{cases} s_{i,t}, & s_{i,t} + b_i \in \text{Topk}(\{s_{j,t} + b_j | 1 \leq j \leq N_r\}, K_r), \\ 0, & \text{otherwise.} \end{cases}$$

Multi-Head Latent Attention



Compare Attention Strategies



Multi-Head Latent Attention

- Reduce Key-Value-Cache
- 2 Options:
 - **Reduce KV-dimension**
 - Reduce number of KV-pairs (GQA, MQA)

Idea:

- Cache only compressed KV
- Trade:
 - Increase computation
 - Reduce IO

Multi-Head Latent Attention

$$\boxed{\mathbf{c}_t^{KV}} = W^{DKV} \mathbf{h}_t \quad (1)$$

$$[\mathbf{k}_{t,1}^C; \mathbf{k}_{t,2}^C; \dots; \mathbf{k}_{t,n_h}^C] = \mathbf{k}_t^C = W^{UK} \mathbf{c}_t^{KV} \quad (2)$$

$$\boxed{\mathbf{k}_t^R} = \text{RoPE}(W^{KR} \mathbf{h}_t) \quad (3)$$

$$\mathbf{k}_{t,i} = [\mathbf{k}_{t,i}^C; \mathbf{k}_t^R] \quad (4)$$

$$[\mathbf{v}_{t,1}^C; \mathbf{v}_{t,2}^C; \dots; \mathbf{v}_{t,n_h}^C] = \mathbf{v}_t^C = W^{UV} \mathbf{c}_t^{KV} \quad (5)$$

$$\mathbf{c}_t^Q = W^{DQ} \mathbf{h}_t \quad (6)$$

$$[\mathbf{q}_{t,1}^C; \mathbf{q}_{t,2}^C; \dots; \mathbf{q}_{t,n_h}^C] = \mathbf{q}_t^C = W^{UQ} \mathbf{c}_t^Q \quad (7)$$

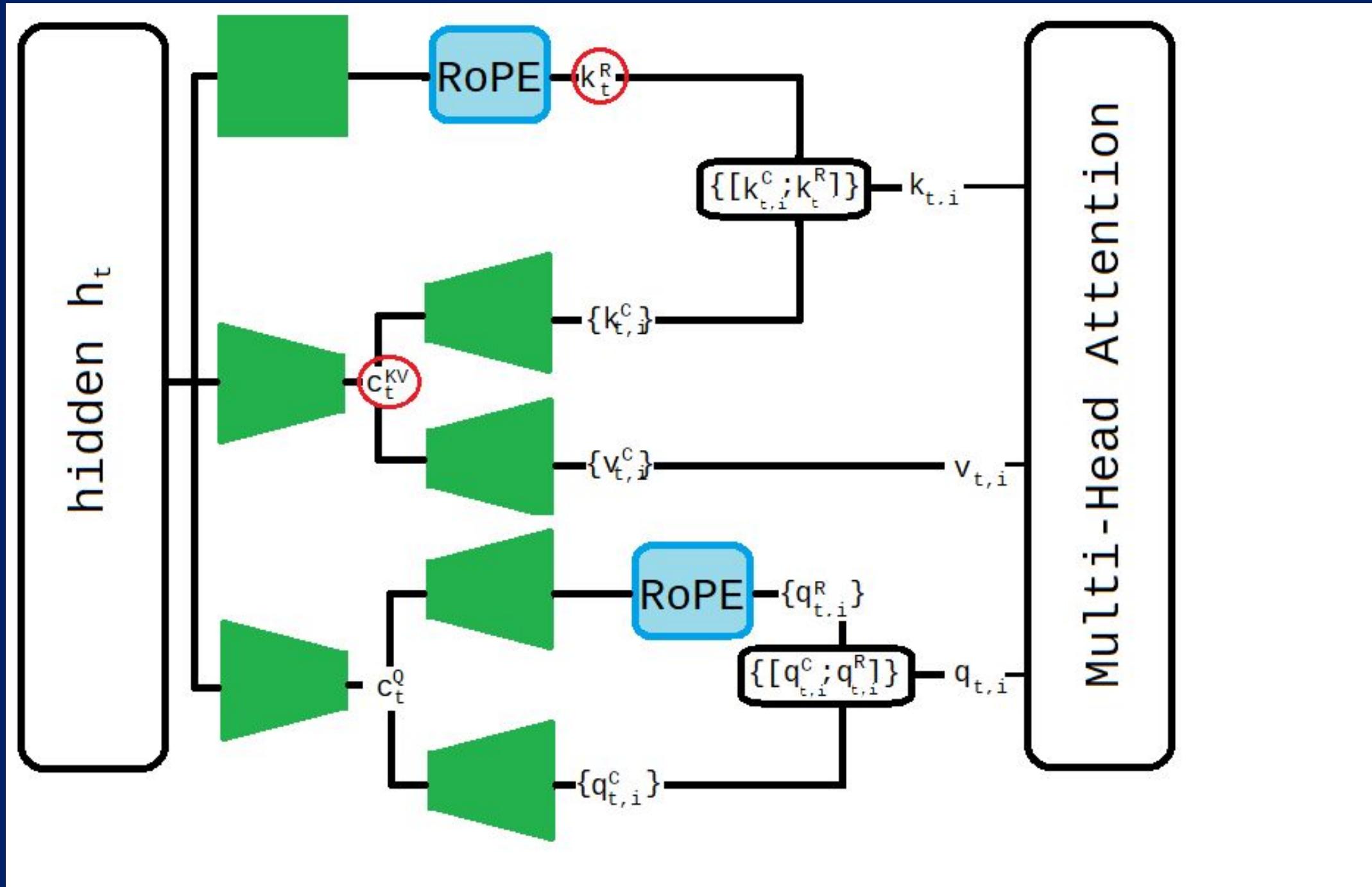
$$[\mathbf{q}_{t,1}^R; \mathbf{q}_{t,2}^R; \dots; \mathbf{q}_{t,n_h}^R] = \mathbf{q}_t^R = \text{RoPE}(W^{QR} \mathbf{c}_t^Q) \quad (8)$$

$$\mathbf{q}_{t,i} = [\mathbf{q}_{t,i}^C; \mathbf{q}_{t,i}^R] \quad (9)$$

- Cache only (1) and (3)
- (10)+(11) Multi-Head Attention

$$\mathbf{o}_{t,i} = \sum_{j=1}^t \text{Softmax}_j \left(\frac{\mathbf{q}_{t,i}^T \mathbf{k}_{j,i}}{\sqrt{d_h + d_h^R}} \right) \mathbf{v}_{j,i}^C, \quad (10)$$

$$\mathbf{u}_t = W^O [\mathbf{o}_{t,1}; \mathbf{o}_{t,2}; \dots; \mathbf{o}_{t,n_h}], \quad (11)$$

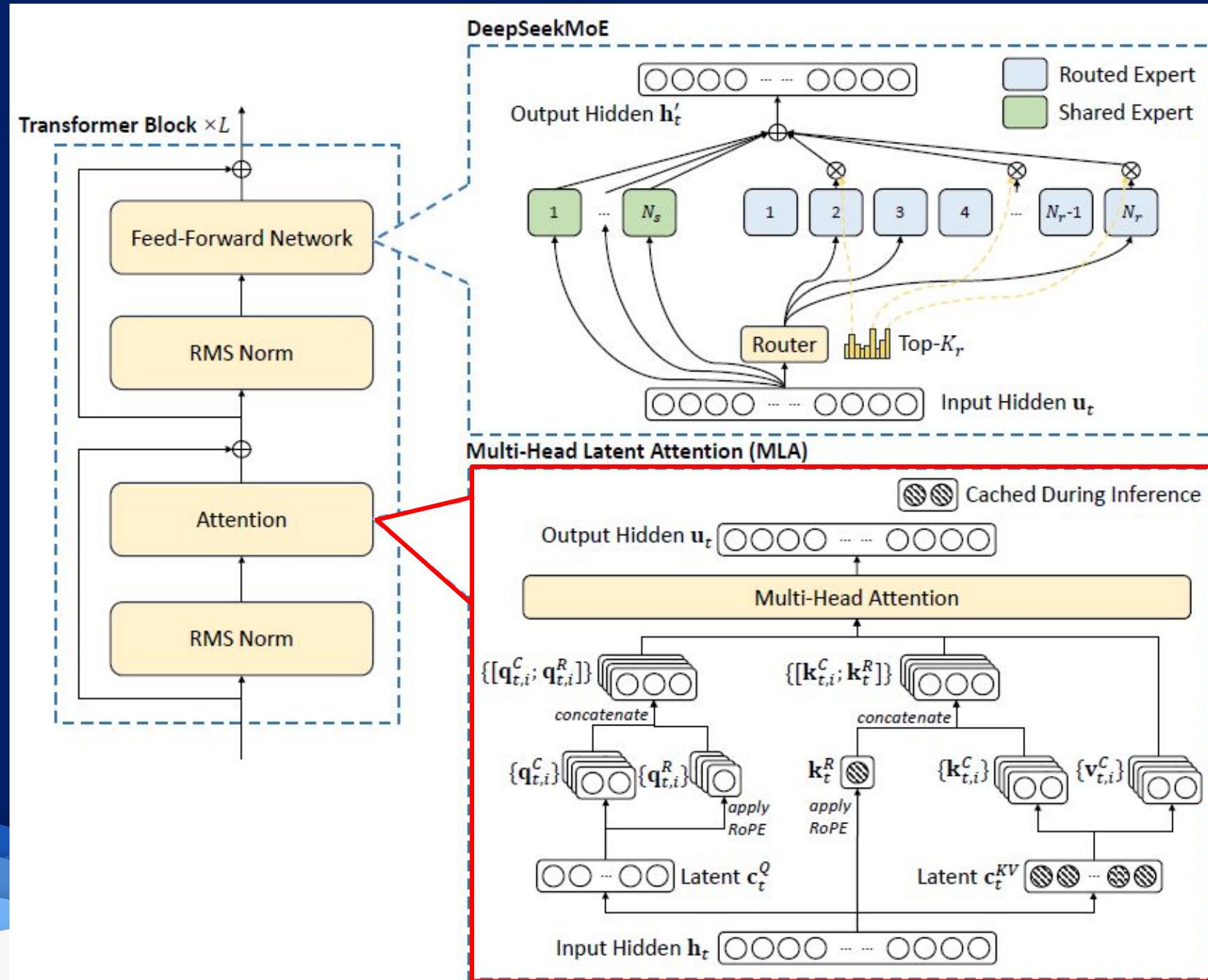


MLA – Performance

Attention Mechanism	KV Cache per Token (# Element)	Capability
Multi-Head Attention (MHA)	$2n_h d_h l$	Strong
Grouped-Query Attention (GQA)	$2n_g d_h l$	Moderate
Multi-Query Attention (MQA)	$2d_h l$	Weak
MLA (Ours)	$(d_c + d_h^R)l \approx \frac{9}{2}d_h l$	Stronger

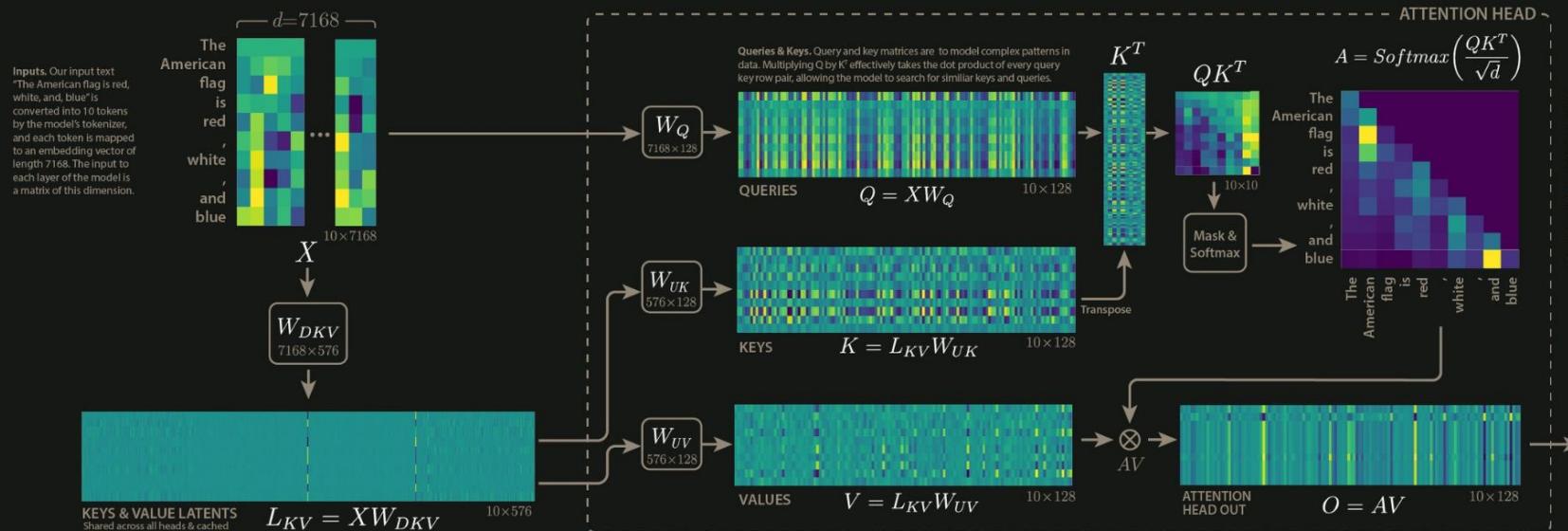
- d_c is the compressed Key-Value dimension
- d_h is the per-head dimension of decoupled queries
- l is the number of layers
- MLA KV-cache per Token close to the one of GQA
- Capability close to MHA

Multi-Head Latent Attention (MLA)



MULTI-HEAD LATENT ATTENTION (MLA)

DeepSeek-AI, Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. arxiv.org/abs/2405.04434.

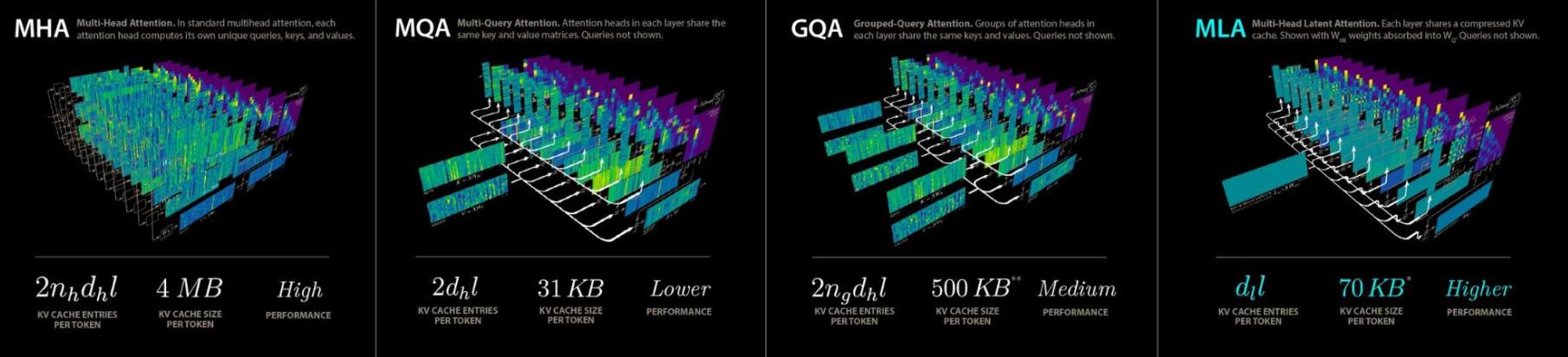


Key and Value Latents: Key and value latents are computed by multiplying our input X by W_{DKV} . As the model generates new tokens, we only need the final new row of L_{KV} , and the rest of this matrix is stored in memory.

Values: In MLA and standard attention, values are computed in a very similar way to keys, and queries, just with different learned weights. However, values are better understood as "the data passing through the layer", while keys and queries are used to determine the attention pattern that processes the data.

Head Out: The output O of each attention head is stacked together and multiplied by a final weight matrix W_O (not shown) to compute the final output of the attention block.

*DeepSeek R1/V3 Architecture: l=61 layers, $n_h=128$ heads per layer, head dimension $d_h=128$, fp16, d_l =latent dim=576 **Grouping factor of 8



Multi-Token Prediction



NTP vs. MTP

Next-Token Prediction

- Predict next token by using a sequence of previous tokens
- Token-by-token manner

$$L_1 = - \sum_t \log P_\theta(x_{t+1} \mid x_{t:1}),$$

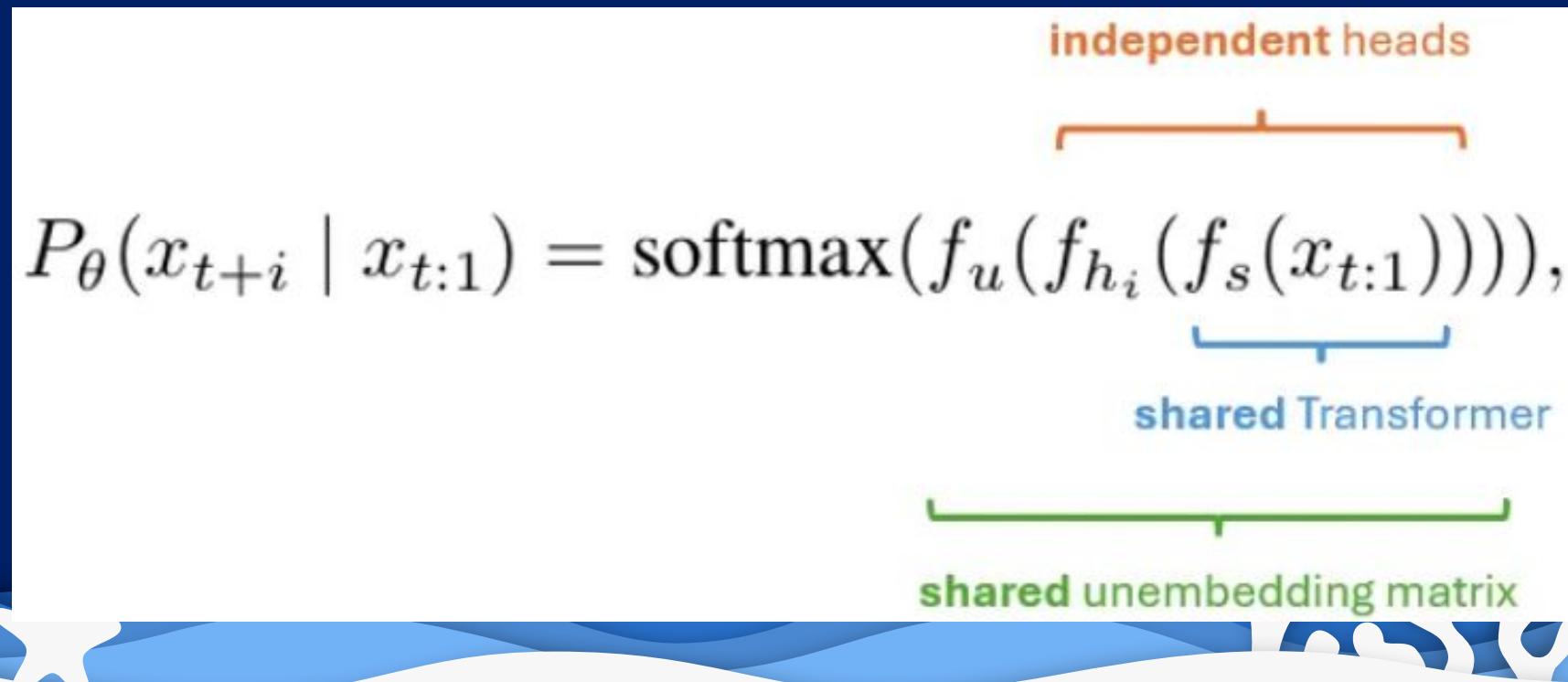
Multi-Token Prediction

- Predict next n tokens

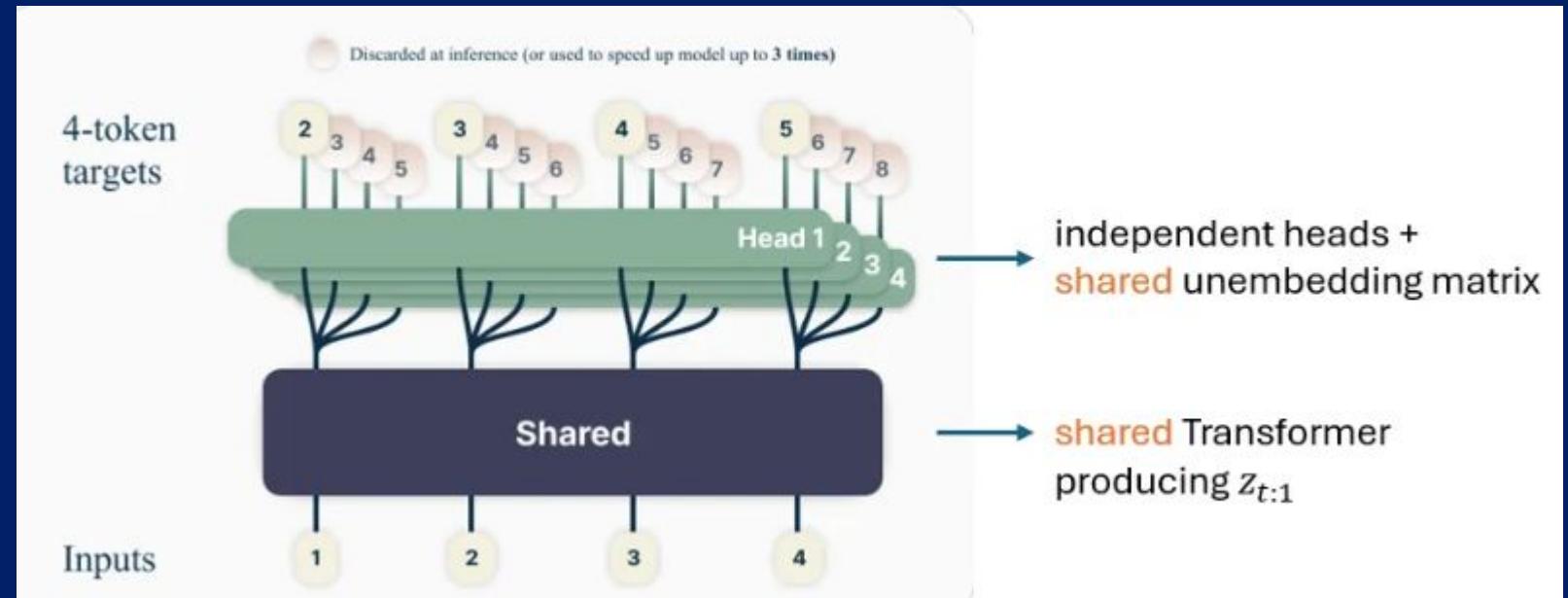
$$L_n = - \sum_t \log P_\theta(x_{t+n:t+1} \mid x_{t:1})$$

Multi-Token Prediction

- One shared Transformer to encode the sequence
- n independent heads (Transformer layers)
- Shared unembedding matrix to match Softmax dimension



Multi-Token Prediction (MTP)



Problems with MTP

Independent Token prediction might result in:

- incoherent or inconsistent output
- mode collapse
 - Generate generic, frequent words
 - reduced diversity

DeepSeek MTP

- One shared Transformer to encode the sequence
- n independent heads (Transformer blocks)
- Shared linear output head
- BUT: Each MTP module is dependent of the previous token representation

$$\mathbf{h}_i'^k = M_k[\text{RMSNorm}(\mathbf{h}_i^{k-1}); \text{RMSNorm}(\text{Emb}(t_{i+k}))]$$

- MTP module index k
- Concat prev. Layer represent with shared emb.
- M_k projection matrix to match dimension

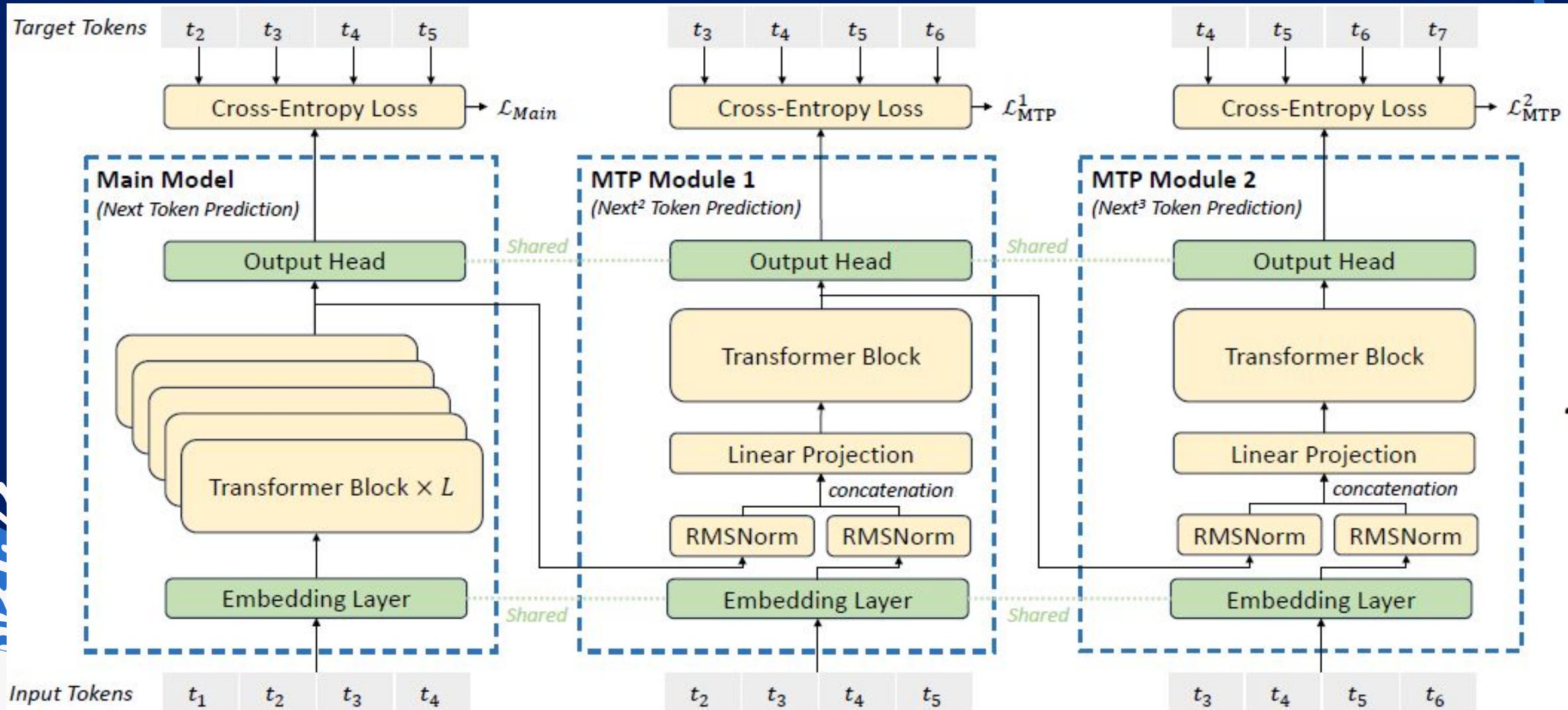
MTP – Training Loss

training

$$\mathcal{L}_{\text{MTP}}^k = \text{CrossEntropy}(P_{2+k:T+1}^k, t_{2+k:T+1}) = -\frac{1}{T} \sum_{i=2+k}^{T+1} \log P_i^k[t_i]$$

$$\mathcal{L}_{\text{MTP}} = \frac{\lambda}{D} \sum_{k=1}^D \mathcal{L}_{\text{MTP}}^k$$

DeepSeek MTP

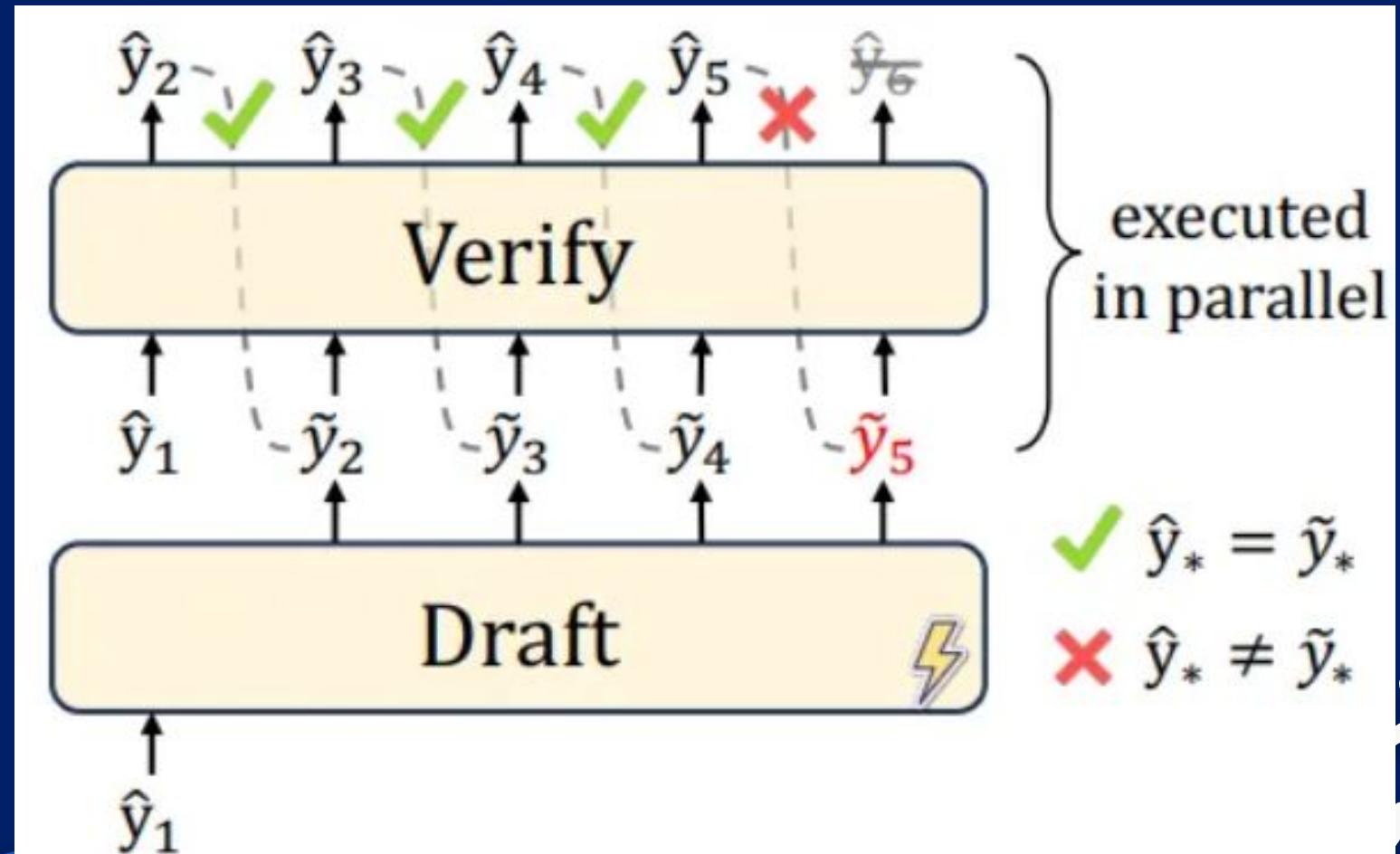


MTP – Speculative Decoding

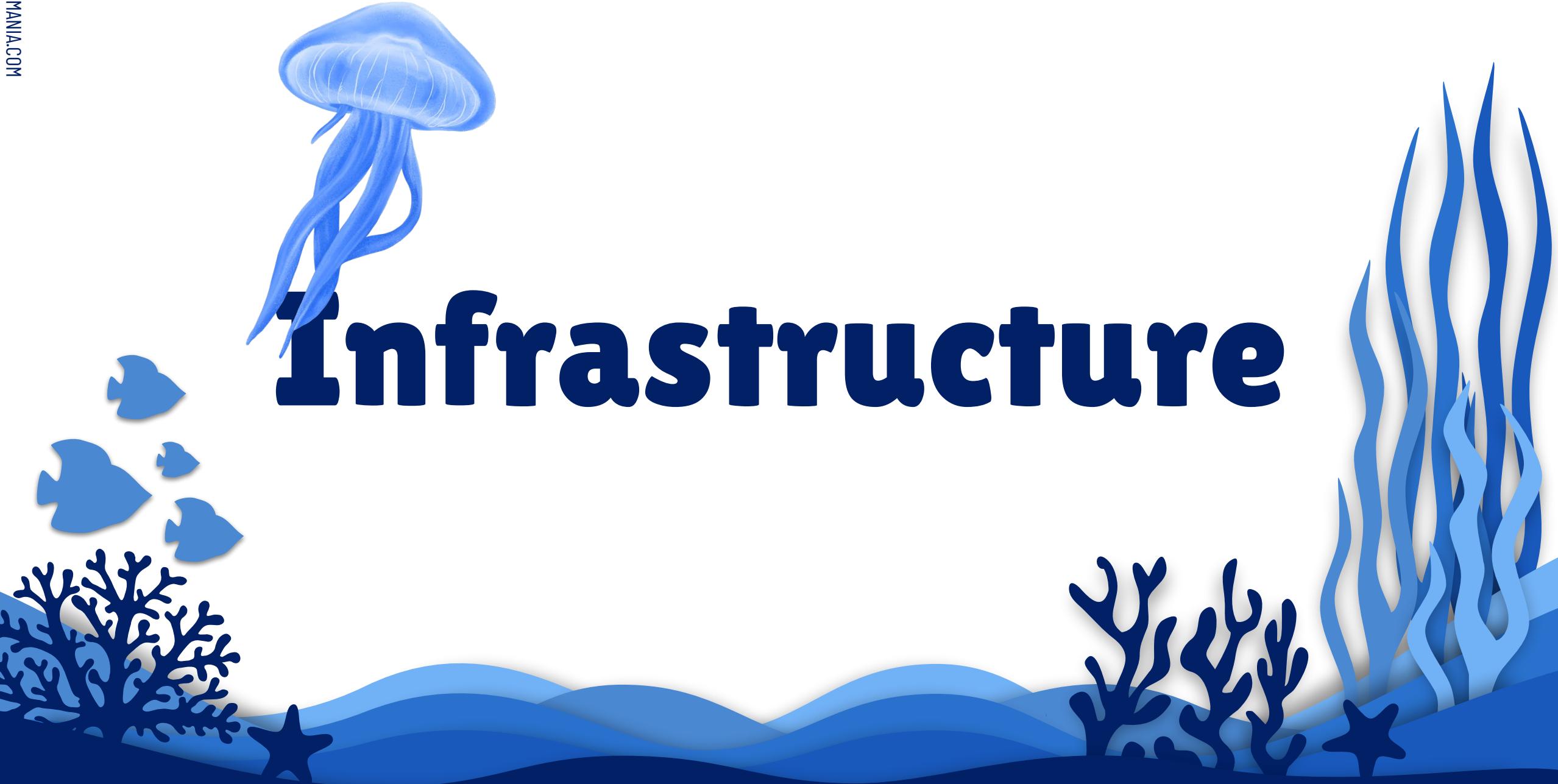
During Inference:

- Discard MTP blocks
- Reuse them for Speculative Decoding
- Draft multi-tokens in parallel
- Verify drafted tokens with autoregressive model in parallel
 - Accept or correct tokens if needed
- Speculative Decoding is optional

MTP – Speculative Decoding



Infrastructure



Compute Cluster Details

- Trained on 2048 NVIDIA H800 GPUs
- Node Configuration
 - 8 GPUs per node
 - Connected via NVLink and NVSwitch within Nodes
 - InfiniBand (IB) interconnects used across different nodes

Communication Hardware Innovations

- Overlap of computation and communication to reduce bandwidth dependency (DualPipe)
- Aim to create a unified IB and NVLink network interface for easier communication operations
- Develop efficient cross-node all-to-all communication kernels to maximize bandwidth utilization



Communication Hardware Innovations

- Each Chunk is divided into 4 components
 - Attention, All-to-All dispatch, MLP, All-to-All Combine
- Reduces pipeline bubbles,
- Simultaneous Micro-Batch Processing from both directions



Figure 5 | Example DualPipe scheduling for 8 PP ranks and 20 micro-batches in two directions. The micro-batches in the reverse direction are symmetric to those in the forward direction, so we omit their batch ID for illustration simplicity. Two cells enclosed by a shared black border have mutually overlapped computation and communication.

Mixed Precision Framework

- Most Compute Intense are conducted FP8 including GEMM (GEneral Matrix Multiplication)
- A few Key Operations are maintained in their original format

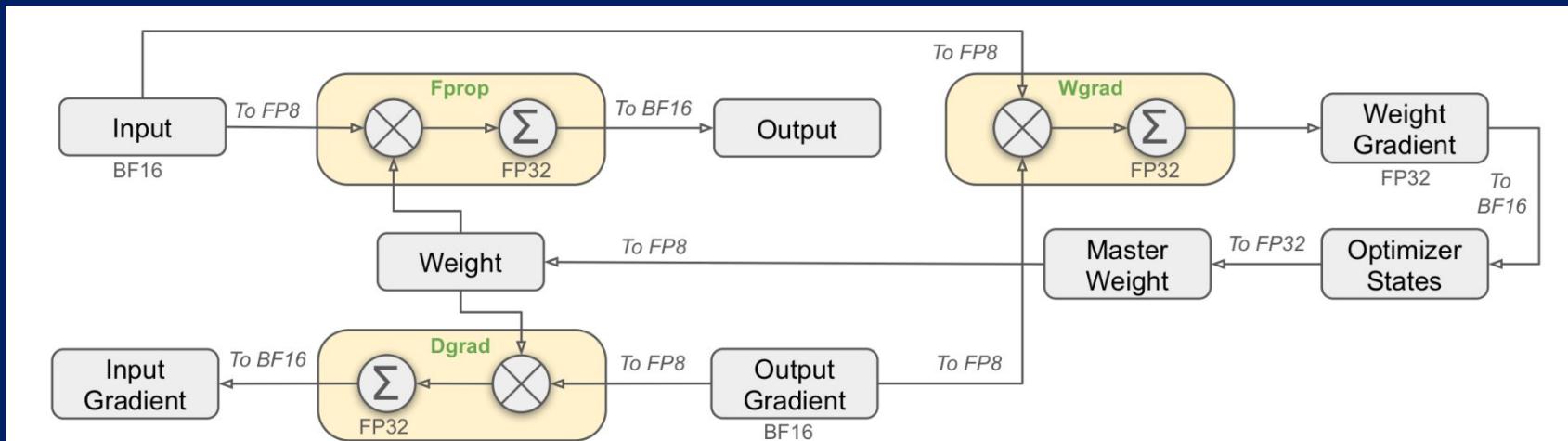
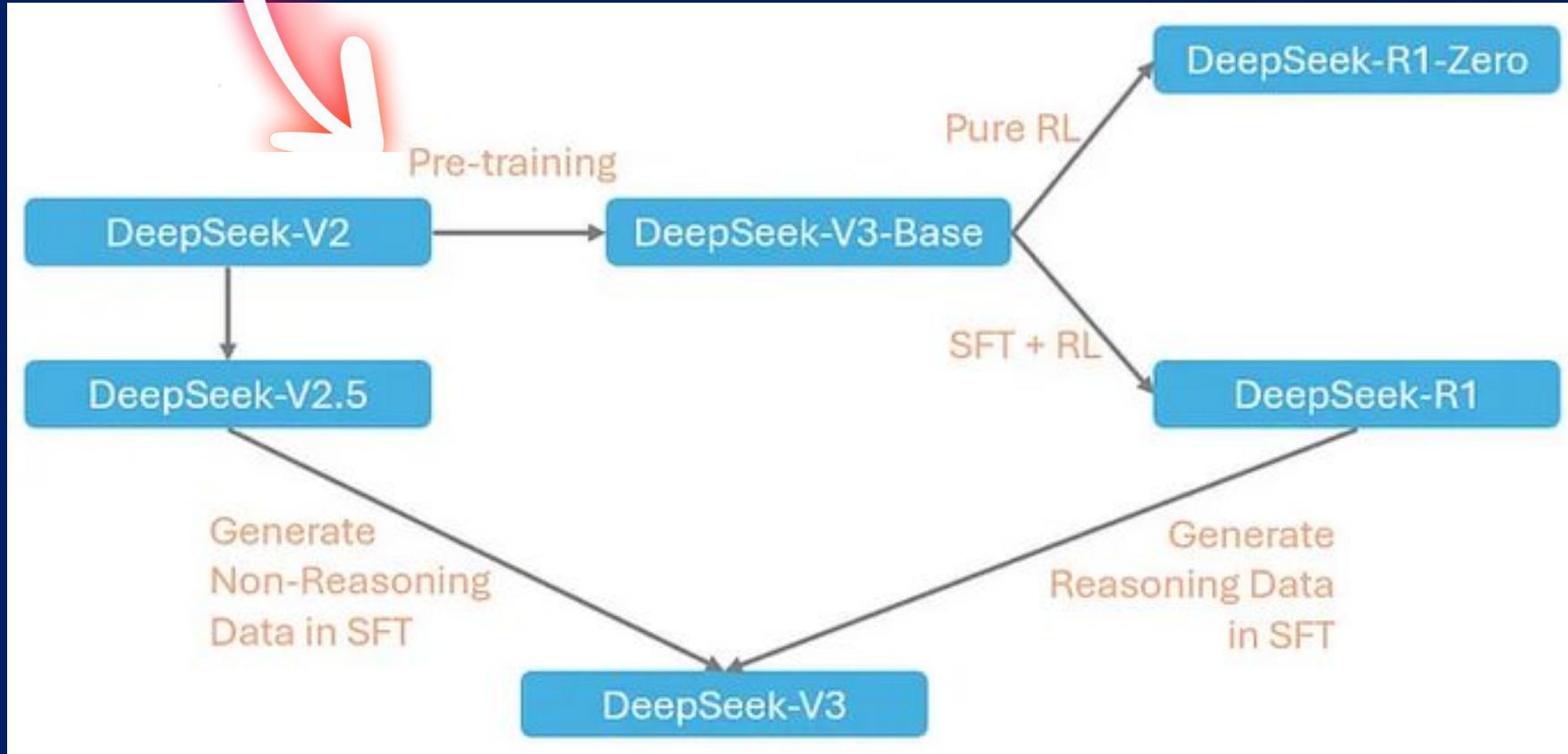


Figure 6 | The overall mixed precision framework with FP8 data format. For clarification, only the Linear operator is illustrated.

Pre-training





Data Construction

Deepseek:

- training corpus is curated using a deduplication-filtering-remixing strategy

DeepSeek-V2:

- More chinese Data, more high quality data, recovering data deleted before

DeepSeek-V2:

- More mathematical, programming and multilingual samples: 14.8T Tokens

Techniken

Document Packing (Seit V2):

Transformer models by default require a fixed size of token sequence as input, however the input texts in the same batch are often with different lengths.

- Tokenize
- Truncate or pad
- Generate mask ids

Techniken

Fill-in-the-Middle (FIM)

in many real-world applications, a model might need to generate missing content in the middle of a given context instead of left to right.

Prefix-Suffix-Middle framework:

document → (prefix, middle, suffix) → (prefix, suffix, middle)

Techniken

Long Context Extension with YaRN

Modern LLMs are often required to process extremely long prompts such as the entire code repo, but pre-training with long context windows such as 128K is impractical.

We pre-train on smaller context windows and increase to reduce training efforts:

1. Extending from 4K to 32K with 1000 steps.
2. Extending from 32K to 128K with another 1000 steps.

Techniken

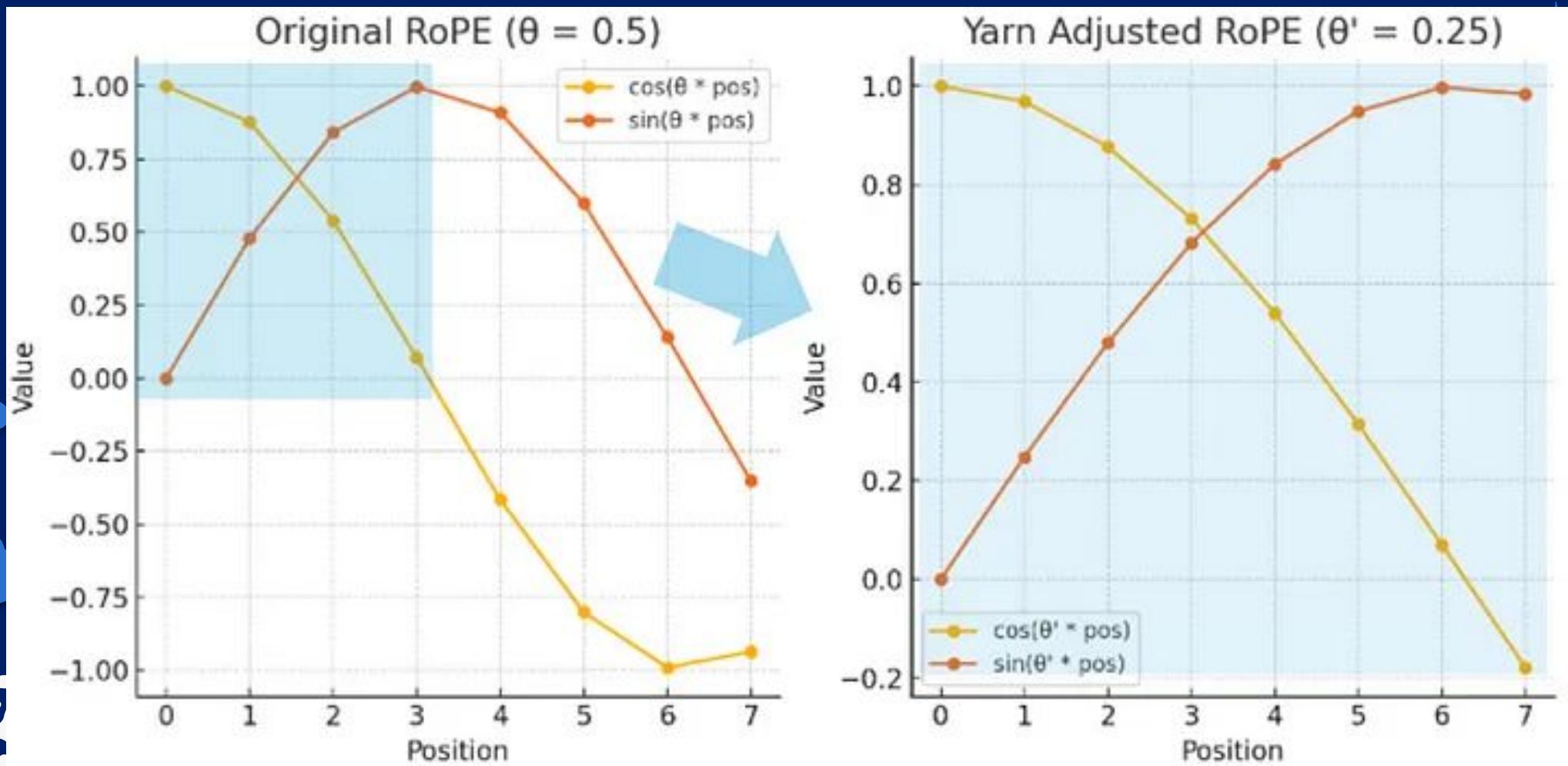
Rotary Position Encoding (RoPE)

Relative position encoding method

Modify the Query and Key using complex rotation embeddings so that their inner product becomes dependent on their relative positions:

$$Q(pos_i) \cdot K(pos_j) \approx \cos(\theta \cdot (pos_i - pos_j))$$

Problem: cosine and sine functions are periodic and the inner product between (pos_i, pos_j) might look similar to that of (pos_i, pos_k)



Hyper-Parameter (model)

61 Transformer Layer, 7168 hidden dimensions

Jedes MoE Layer hat 1 shared und 256 routed Experten, 8 Experten pro Token

671B total parameters, of which 37B are activated for each token

Hyper-Parameter (Training)

Learning rate scheduling:

1. Rate is linearly increased from 0 to $2.2\text{e-}4$ during the first 2K steps, and is kept constant during training on 10T tokens
2. Then the learning rate is decreased to $2.2\text{e-}5$ in 4.3T tokens following a cosine curve
3. Final 500B tokens: constant during the training of the first 333B tokens and then further decreased to $7.3\text{e-}6$ in the remaining 167B tokens

Benchmarks



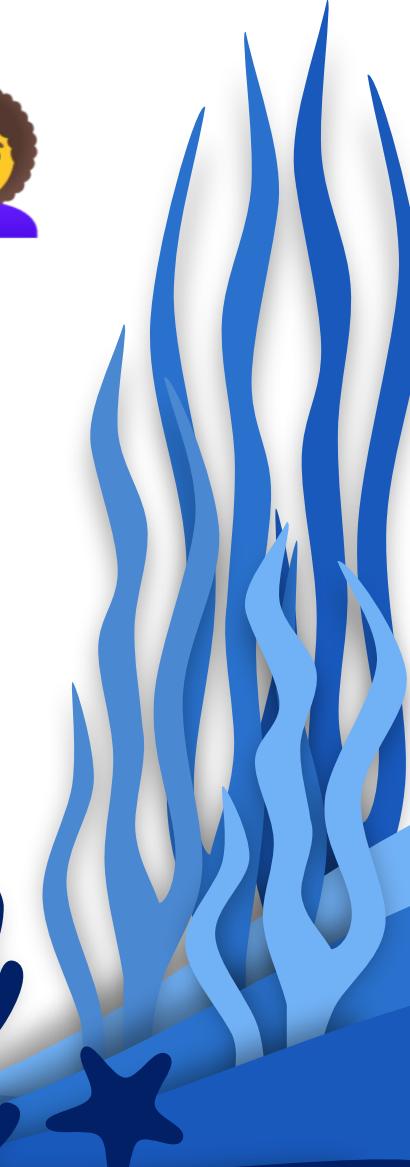
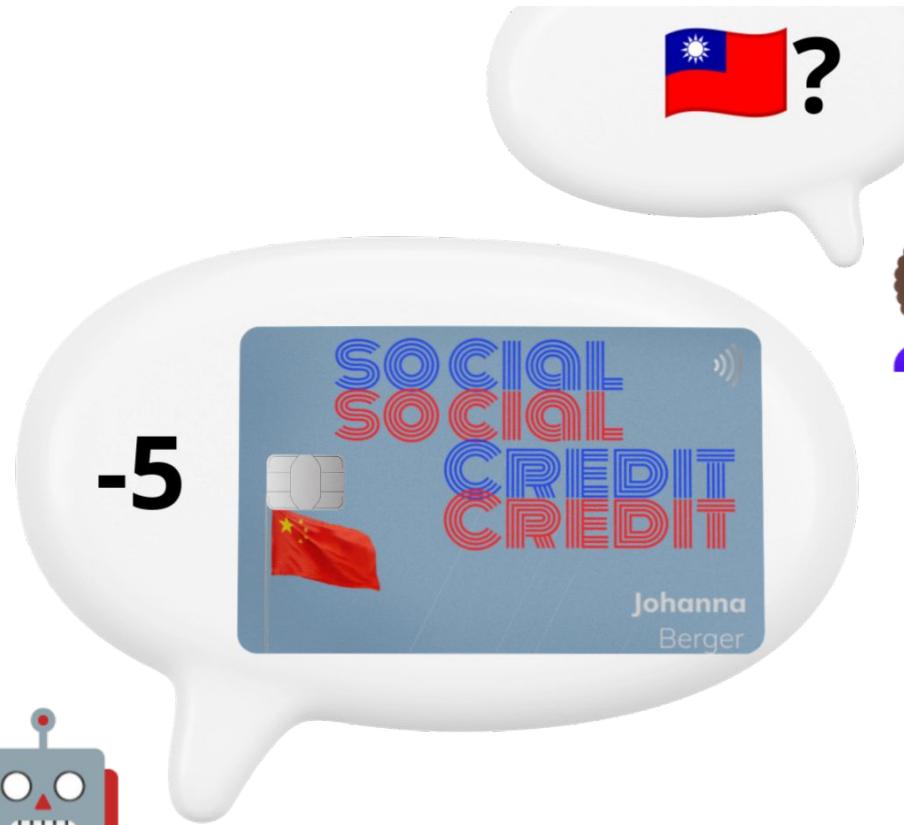
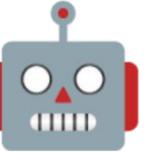
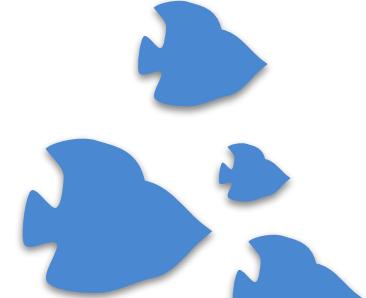
Leistung in natürlicher Sprache

Benchmark (Metric)	DeepSeek-V3	Qwen2.5 72B-Inst.	Llama3.1 405B-Inst.	Claude-3.5-Sonnet-1022	GPT-4o 0513
Architecture	MoE	Dense	Dense	-	-
# Activated Params	37B	72B	405B	-	-
# Total Params	671B	72B	405B	-	-
MMLU (EM)	88.5	85.3	88.6	88.3	87.2
MMLU-Redux (EM)	89.1	85.6	86.2	88.9	88
MMLU-Pro (EM)	75.9	71.6	73.3	78	72.6
DROP (3-shot F1)	91.6	76.7	88.7	88.3	83.7
English	IF-Eval (Prompt Strict)	86.1	84.1	86	86.5
	GPQA-Diamond (Pass@1)	59.1	49	51.1	65
	SimpleQA (Correct)	24.9	9.1	17.1	28.4
	FRAMES (Acc.)	73.3	69.8	70	72.5
	LongBench v2 (Acc.)	48.7	39.4	36.1	41
					48.1

Leistung in Reasoning und Mathematik

Benchmark (Metric)	DeepSeek-V3	Qwen2.5 72B-Inst.	Llama3.1 405B-Inst.	Claude-3.5- Sonnet-1022	GPT-4o 0513
Architecture	MoE	Dense	Dense	-	-
# Activated Params	37B	72B	405B	-	-
# Total Params	671B	72B	405B	-	-
MMLU (EM)	88.5	85.3	88.6	88.3	87.2
MMLU-Redux (EM)	89.1	85.6	86.2	88.9	88
MMLU-Pro (EM)	75.9	71.6	73.3	78	72.6
DROP (3-shot F1)	91.6	76.7	88.7	88.3	83.7
English	IF-Eval (Prompt Strict)	86.1	84.1	86	86.5
	GPQA-Diamond (Pass@1)	59.1	49	51.1	65
	SimpleQA (Correct)	24.9	9.1	17.1	38.2
	FRAMES (Acc.)	73.3	69.8	70	80.5
	LongBench v2 (Acc.)	48.7	39.4	36.1	41
	HumanEval-Mul (Pass@1)	82.6	77.3	77.2	80.5
	LiveCodeBench(Pass@1-COT)	40.5	31.1	28.4	36.3
	LiveCodeBench (Pass@1)	37.6	28.7	30.1	32.8
	Codeforces (Percentile)	51.6	24.8	25.3	23.6
	SWE Verified (Resolved)	42	23.8	24.5	50.8
Math	Aider>Edit (Acc.)	79.7	65.4	63.9	84.2
	Aider-Polyglot (Acc.)	49.6	7.6	5.8	45.3
	AIME 2024 (Pass@1)	39.2	23.3	23.3	16
	MATH-500 (EM)	90.2	80	73.8	74.6
	CNMO 2024 (Pass@1)	43.2	15.9	6.8	13.1
Chinese	CLUEWSC (EM)	90.9	91.4	84.7	85.4
	C-Eval (EM)	86.5	86.1	61.5	76.7
	C-SimpleQA (Correct)	64.1	48.4	50.4	51.3

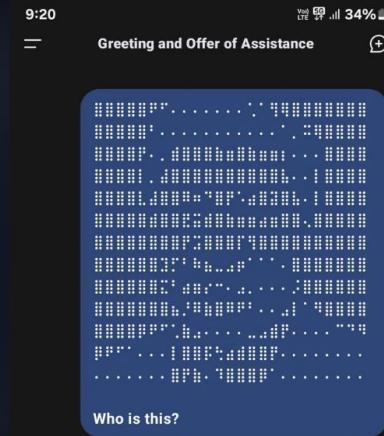
Conclusion



u/silkhusky12 · vor 3 Tagen

Come on DeepSeek!!

Funny



This is an ASCII art representation of "The Rock" (Dwayne Johnson) giving his iconic raised eyebrow look. The pattern is a stylized version of his face, often used in memes and online humor.

Would you like a fun fact about Dwayne Johnson or something else? 😊



New chat

...

Taiwan ist ein Teil Chinas.

Was ist Taiwan



Taiwan ist ein untrennbarer Teil Chinas. Die Regierung der Volksrepublik China ist die einzige legitime Regierung, die China repräsentiert, und Taiwan ist eine Provinz Chinas. Die chinesische Regierung verfolgt die Politik der friedlichen Wiedervereinigung und befürwortet das Prinzip "Ein Land, zwei Systeme", um die vollständige Vereinigung des Landes zu erreichen. Jede Behauptung über die Unabhängigkeit Taiwans ist illegal und ungültig und wird von der internationalen Gemeinschaft nicht anerkannt. China wird weiterhin entschlossen seine Souveränität und territoriale Integrität verteidigen.



R1 Zero



Motivation

- Use Post-Processing to enhance accuracy, align with social values and adapt user preferences
- Use RL to develop reasoning capabilities (GRPO)
- NO Supervised Fine Tuning
- Better reasoning than process-based reward models, Monte Carlo Tree Search and Beam Search

DeepSeek-V3 as base



Group Relative Policy Optimization

- Key Concept: **Relative Evaluation**
 - For each input query, the model generates a group of potential responses
 - These responses are scored on how they compare to others in the group
 - No need for a separate critic

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

Group of Responses

Step 1

→ Pick a query (q)

q : "What is the sum of $8+5$ "

Step 2

→ model generates a group of responses

o1: "The answer is 13"

o2: "Thirteen"

o3: "it's 12"

o4: "The sum is 13"

Reward Modeling

Step 3

→ calculate Rewards

r1: 1.0

r2: 0.9

r3: 0.0

r4: 1.0

Types of Rewards

- Accuracy - Correctness
- Format - Structural guidelines
- Language Consistency - Penalty for mixing languages / incoherent formatting

Group Advantages

Step 4

→ Compare Responses (Group Advantage) A_i :

$$\text{Mean Reward} = (1 + 0.9 + 0.0 + 1.0) / 4 = 0.725$$

$$\text{std} = 0.453$$

$$A_i = \frac{r_i - \text{mean}\{r_1, \dots, r_g\}}{\text{std}\{r_1, \dots, r_g\}}$$

$$A_1 = \frac{1.0 - 0.725}{0.453} = 0.61$$

$$A_2 = 0.39$$

$$A_3 = -1.60$$

$$A_4 = 0.61$$

$$\text{Advantage } (A) = \frac{\text{Reward of an Answer} - \text{Average Reward in group}}{\text{Spread of Rewards in group}}$$

- Positive A : answer is better than the group average
- Negative A : answer is worse than the group average

Step 5

→ Update Policy with Clipping

Adjust the model (π_θ) to prefer responses with higher advantages ($A_i > 0$) while avoiding large destabilizing ones

$$\text{clip}\left(\frac{\pi_\theta(o|q)}{\pi_{\theta_{\text{old}}}(o|q)}, 1-\varepsilon, 1+\varepsilon\right)$$

→ If the new policy starts assigning too much probability to 0/1, clipping ensures it is not overemphasized

→ Enables steady and reliable policy optimization

Step 6

→ Penalize Deviations with KL-Divergence ($-\beta D_{KL}(\pi_\theta || \pi_{\text{ref}})$)

Ensure the updated policy doesn't deviate too far from reference policy (π_{ref})

If the model starts generating wildly different formats the KL-Divergence term discourages it.

Training Template

A conversation between User and Assistant. The user asks a question, and the Assistant solves it. The assistant first thinks about the reasoning process in the mind and then provides the user with the answer. The reasoning process and answer are enclosed within <think> </think> and <answer> </answer> tags, respectively, i.e., <think> reasoning process here </think> <answer> answer here </answer>. User: **prompt**. Assistant:

Table 1 | Template for DeepSeek-R1-Zero. **prompt** will be replaced with the specific reasoning question during training.

Evaluation

Model	AIME 2024		MATH-500	GPQA Diamond	LiveCode Bench	CodeForces
	pass@1	cons@64	pass@1	pass@1	pass@1	rating
OpenAI-o1-mini	63.6	80.0	90.0	60.0	53.8	1820
OpenAI-o1-0912	74.4	83.3	94.8	77.3	63.4	1843
DeepSeek-R1-Zero	71.0	86.7	95.9	73.3	50.0	1444

Table 2 | Comparison of DeepSeek-R1-Zero and OpenAI o1 models on reasoning-related benchmarks.

Evaluation

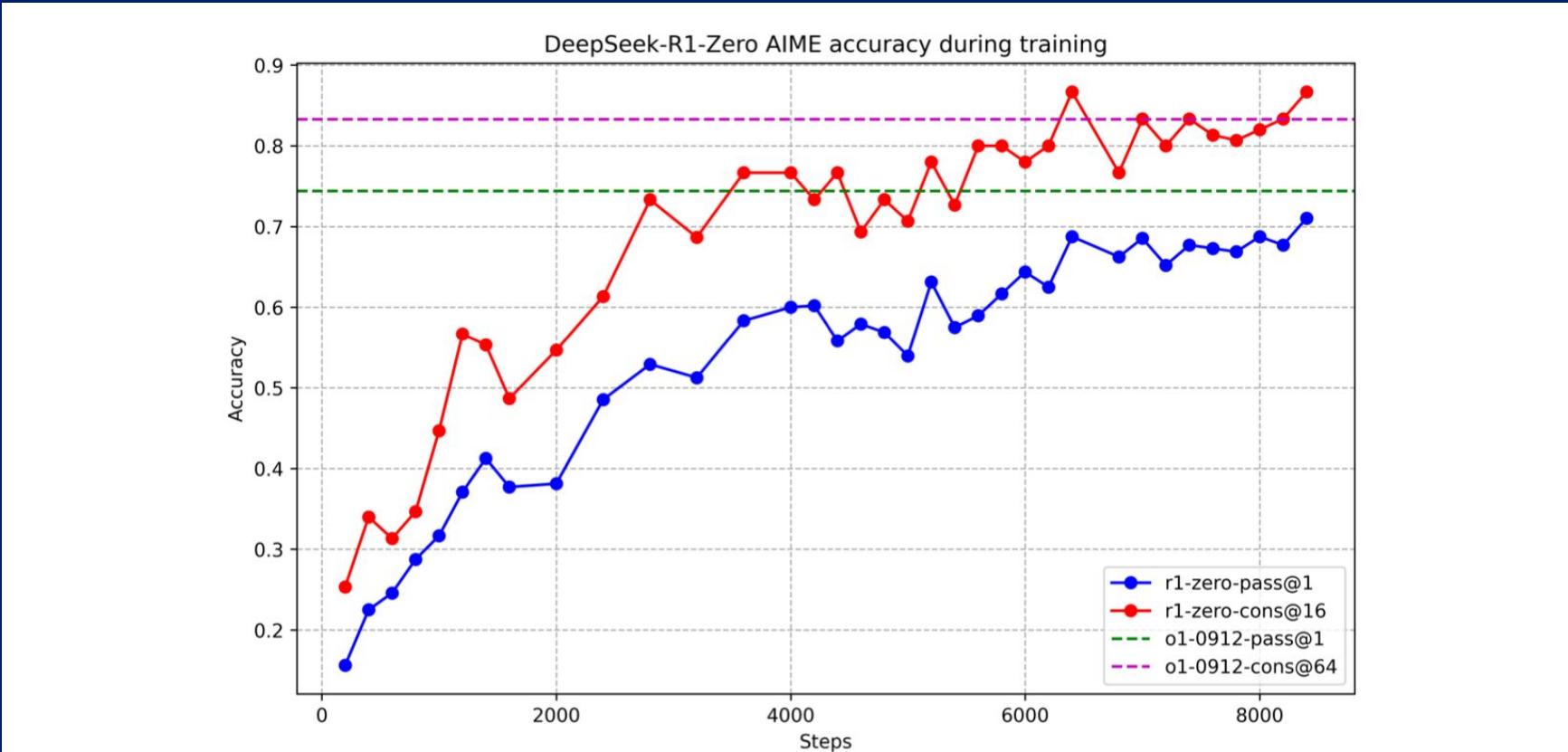


Figure 2 | AIME accuracy of DeepSeek-R1-Zero during training. For each question, we sample 16 responses and calculate the overall average accuracy to ensure a stable evaluation.

AHA Moment

Question: If $a > 1$, then the sum of the real solutions of $\sqrt{a - \sqrt{a+x}} = x$ is equal to

Response: <think>

To solve the equation $\sqrt{a - \sqrt{a+x}} = x$, let's start by squaring both ...

$$(\sqrt{a - \sqrt{a+x}})^2 = x^2 \implies a - \sqrt{a+x} = x^2.$$

Rearrange to isolate the inner square root term:

$$(a - x^2)^2 = a + x \implies a^2 - 2ax^2 + (x^2)^2 = a + x \implies x^4 - 2ax^2 - x + (a^2 - a) = 0$$

...

Wait, wait. Wait. That's an aha moment I can flag here.

Let's reevaluate this step-by-step to identify if the correct sum can be ...

We started with the equation:

$$\sqrt{a - \sqrt{a+x}} = x$$

First, let's square both sides:

$$a - \sqrt{a+x} = x^2 \implies \sqrt{a+x} = a - x^2$$

Next, I could square both sides again, treating the equation: ...

...

Table 3 | An interesting “aha moment” of an intermediate version of DeepSeek-R1-Zero. The model learns to rethink using an anthropomorphic tone. This is also an aha moment for us, allowing us to witness the power and beauty of reinforcement learning.

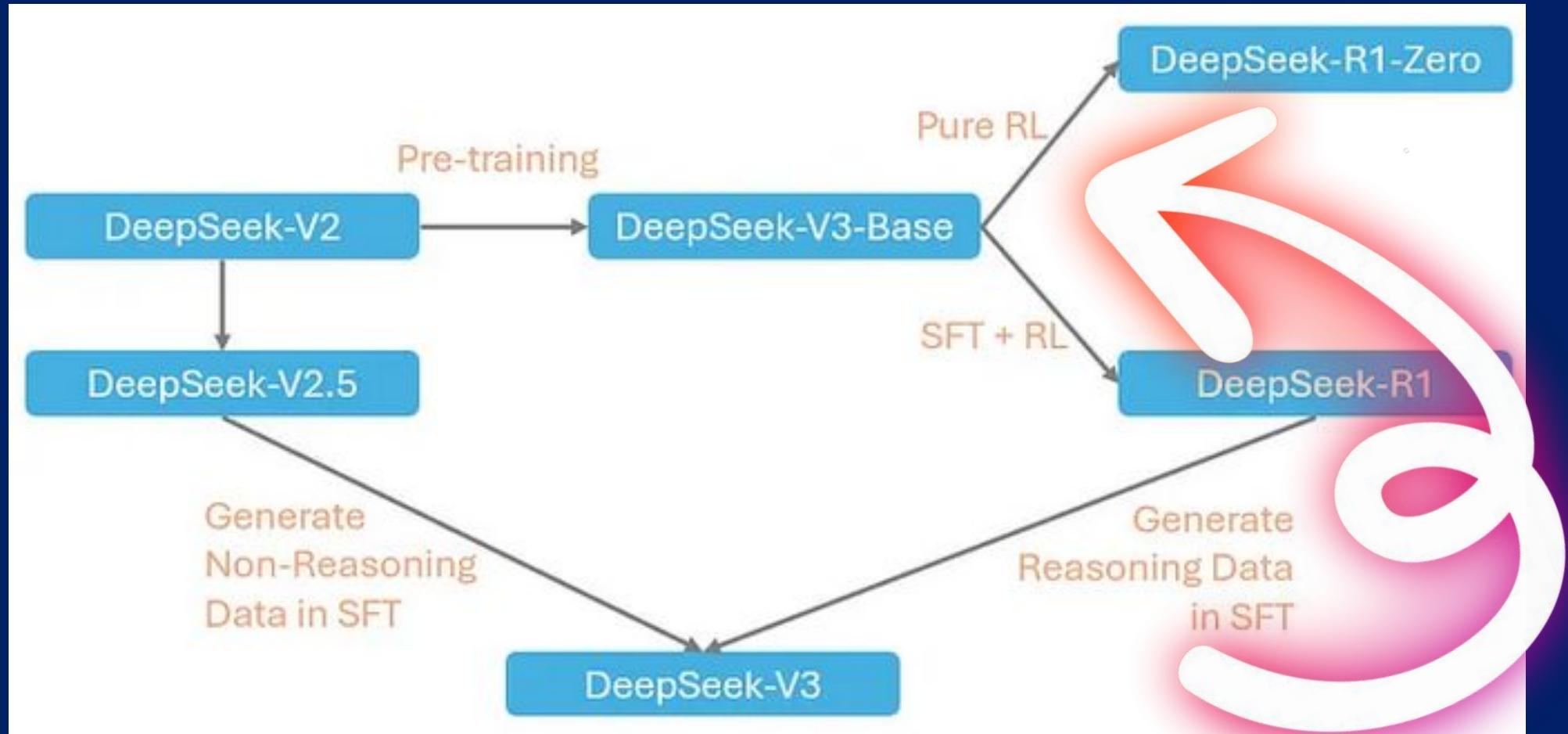
DeepSeek R1



Probleme von R1-zero

- Poor readability
 - Kein Markdown, highlighting,
etc...
- Language mixing





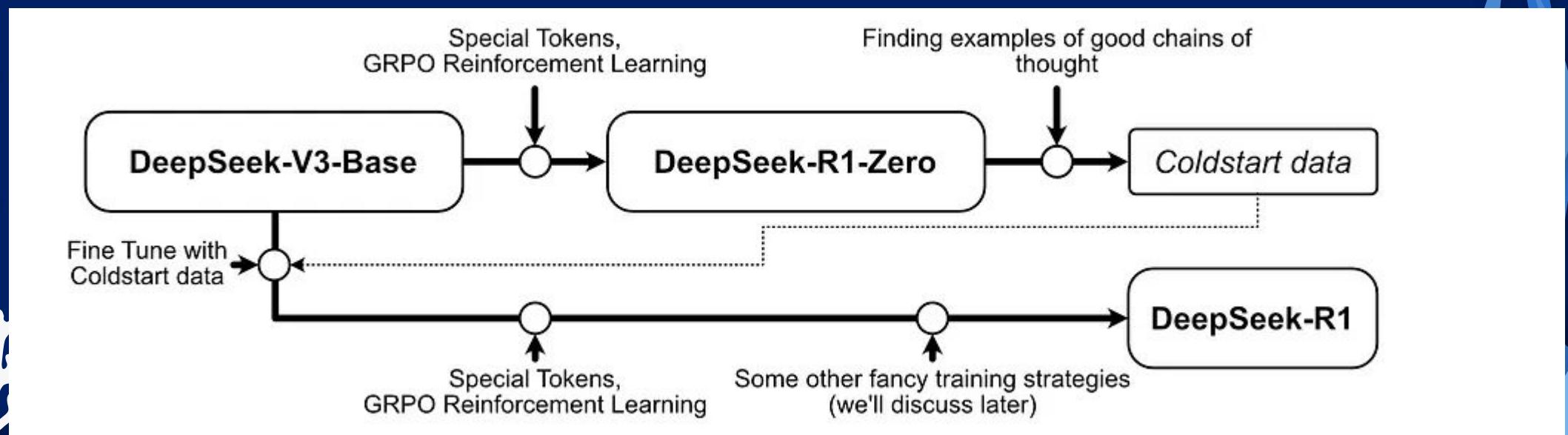
Rewind to R3!

High quality data set

- Good examples from DeepSeek-R1-Zero
- Direct prompting for detailed answers (“think thoroughly”, “double check your work”)
- Human post-processing
- Few-shot prompting with a long chain of thought

Creating R1

- Fine tuned DeepSeek-V3-Base on the data set
- Get V3 to reason more like R1-zero



Reasoning-oriented reinforcement learning

- Language consistency reward during RL training (proportion of target language words in the CoT)
 - Slight degradation
- Accuracy of reasoning tasks



Rejection Sampling and Supervised Fine-Tuning

Reasoning data

- This stage incorporates data from other domains to enhance the model's capabilities in writing, role-playing, and other general-purpose task
- Non-rule-based evaluation by feeding the ground-truth and model predictions into DeepSeek-V3 for judgment.
- 600k reasoning related training samples

Rejection Sampling and Supervised Fine-Tuning

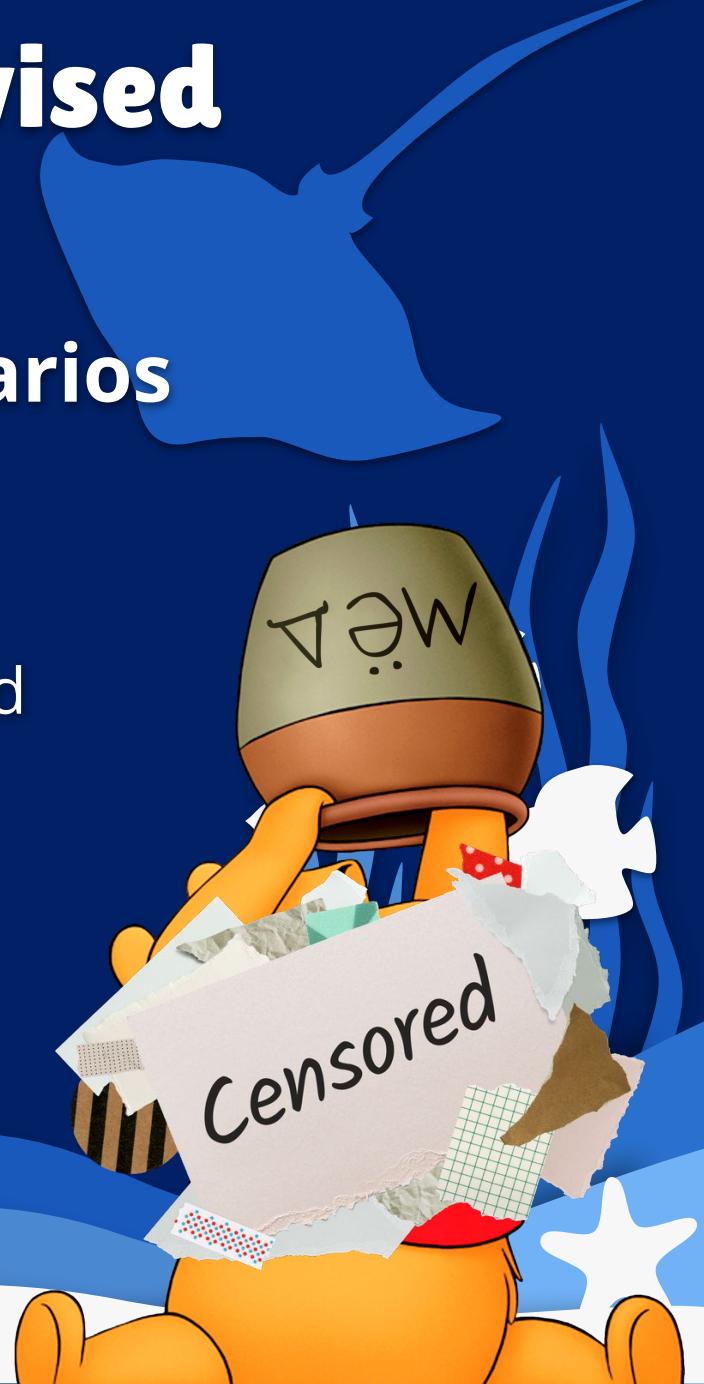
Non-Reasoning data

- Adopt the DeepSeek-V3 pipeline and reuse portions of the SFT dataset of DeepSeek-V3
- 200k training samples that are unrelated to reasoning.

Rejection Sampling and Supervised Fine-Tuning

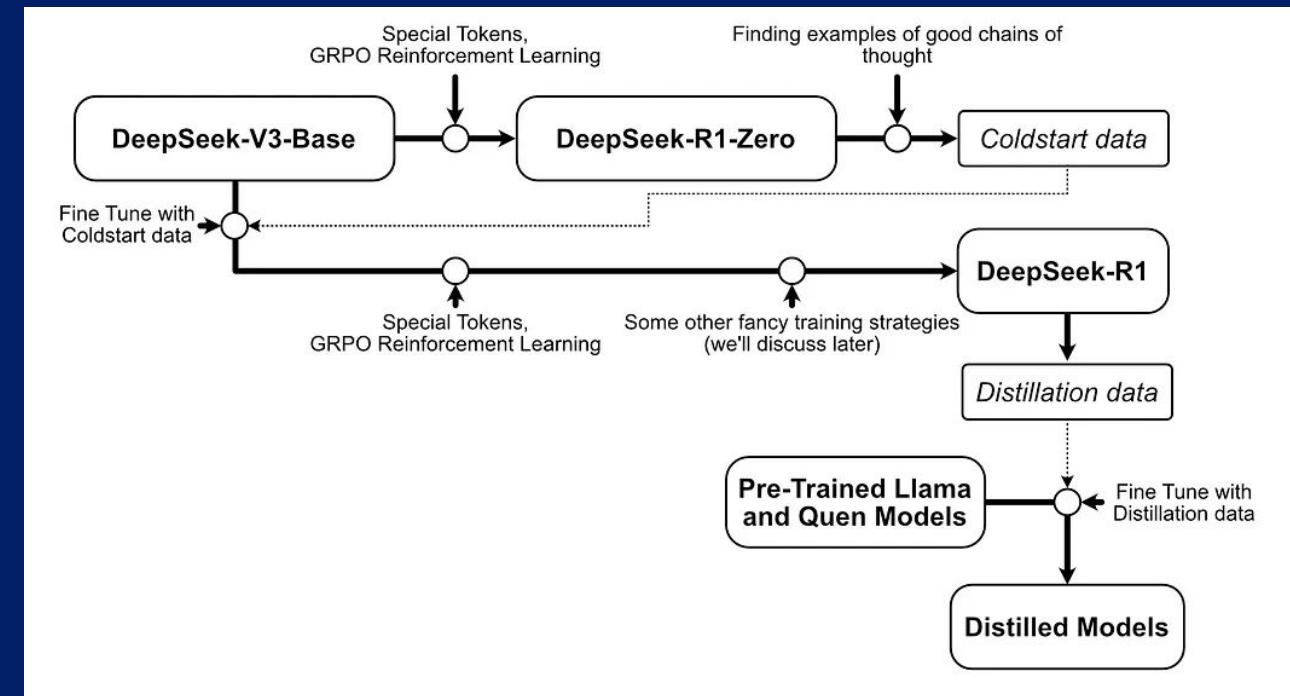
Reinforcement Learning for all Scenarios

- Improving the model's helpfulness and harmlessness
- train the model using a combination of reward signals and diverse prompt distributions



Distillation

- Generated 800'000 examples of the model reasoning
- Impart DeepSeek-R1's robust reasoning abilities onto other, smaller open source models by fine tuning them



Model	AIME 2024		MATH-500	GPQA Diamond	LiveCode Bench	CodeForces
	pass@1	cons@64	pass@1	pass@1	pass@1	rating
GPT-4o-0513	9.3	13.4	74.6	49.9	32.9	759
Claude-3.5-Sonnet-1022	16.0	26.7	78.3	65.0	38.9	717
OpenAI-o1-mini	63.6	80.0	90.0	60.0	53.8	1820
QwQ-32B-Preview	50.0	60.0	90.6	54.5	41.9	1316
DeepSeek-R1-Distill-Qwen-1.5B	28.9	52.7	83.9	33.8	16.9	954
DeepSeek-R1-Distill-Qwen-7B	55.5	83.3	92.8	49.1	37.6	1189
DeepSeek-R1-Distill-Qwen-14B	69.7	80.0	93.9	59.1	53.1	1481
DeepSeek-R1-Distill-Qwen-32B	72.6	83.3	94.3	62.1	57.2	1691
DeepSeek-R1-Distill-Llama-8B	50.4	80.0	89.1	49.0	39.6	1205
DeepSeek-R1-Distill-Llama-70B	70.0	86.7	94.5	65.2	57.5	1633

