DeepSeek-V3 Technical Report

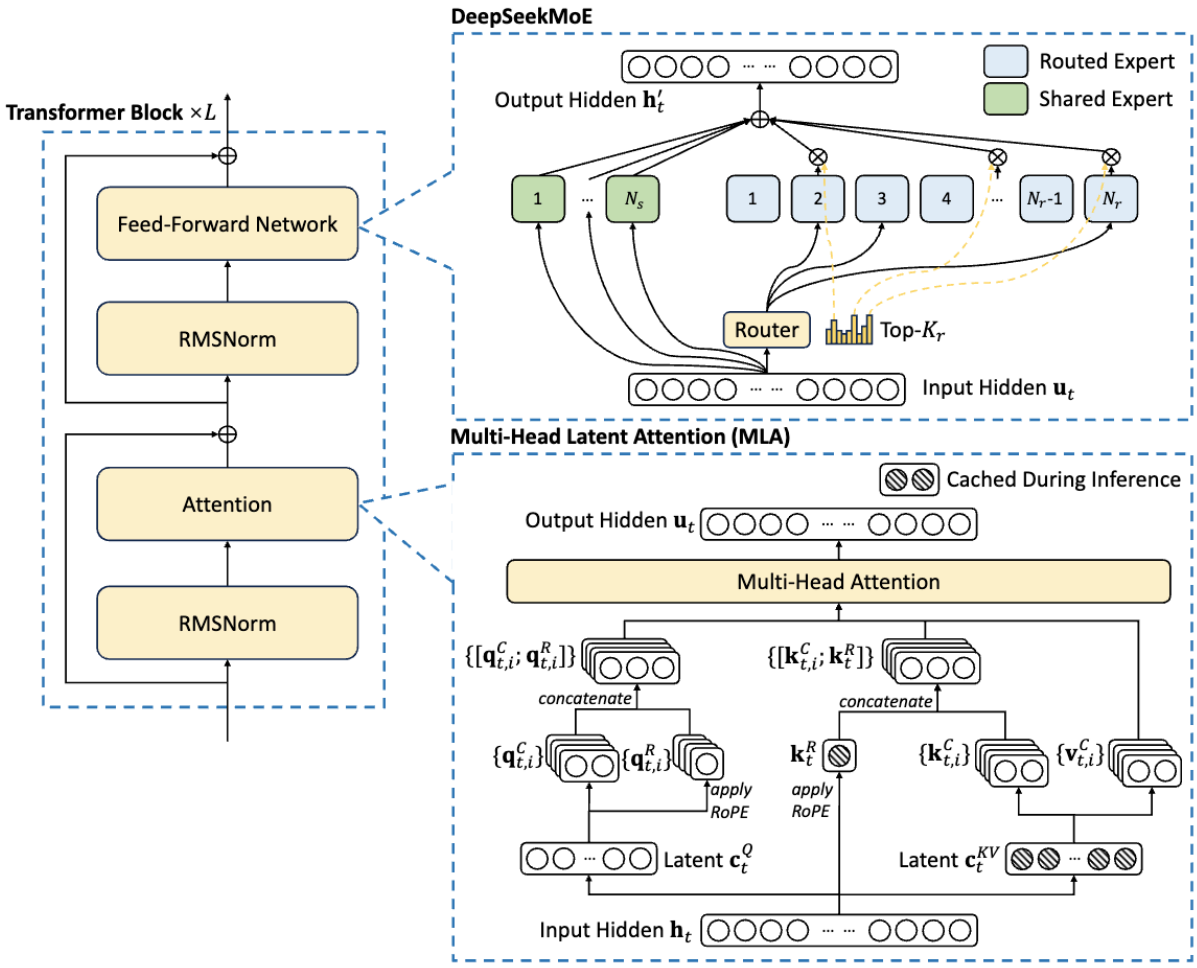
- https://arxiv.org/pdf/2412.19437

概要

- DeepSeek-V3の提案と学習方法記載
- DeepSeek-V3はTransformerベースのLLMでMulti-head Latent Attention(MLA)とMixture-of-Experts(MoE)を採用したもの
- 超軽量なのにもかかわらずgpt4o, claude-3.5を上回る性能
- 事後学習はCoT(Chain-of-Thought)モデルのDeepSeek R1から蒸留

全体図

- 以下全体図



Multi-head Latent Attention (MLA)

- 上の図がわかりやすい
- key, valueは以下のようにして次元の小さい潜在変数に変換してから復元したものを使ってattentionを計算する

$$
\begin{aligned}
\boxed{\mathbf{c}_t^{KV}} &= W^{DKV}\mathbf{h}_t, \\
[\mathbf{k}_{t,1}^C; \mathbf{k}_{t,2}^C; \ldots; \mathbf{k}_{t,n_h}^C] = \mathbf{k}_t^C &= W^{UK}\mathbf{c}_t^{KV}, \\
\boxed{\mathbf{k}_t^R} &= \mathrm{RoPE}(W^{KR}\mathbf{h}_t), \\
\mathbf{k}_{t,i} &= [\mathbf{k}_{t,i}^C; \mathbf{k}_t^R], \\
[\mathbf{v}_{t,1}^C; \mathbf{v}_{t,2}^C; \ldots; \mathbf{v}_{t,n_h}^C] = \mathbf{v}_t^C &= W^{UV}\mathbf{c}_t^{KV},
\end{aligned}
$$

- このときkv cacheは青字の潜在次元の変数だけをキャッシュしておけばよくメモリ削減になる
- $h_t \in \mathbb{R}^d$: t番目のtokenに対応する入力
- n_h : the number of attention heads
- d_h : dimension per head
- $c_t^{kv} \in \mathbb{R}^{d_c}$: compressed latent vector for keys and values ($d_c \ll d_h n_h$)
- d_c : KV compression dimension
- $W^{DKV} \in \mathbb{R}^{d_c \times d}$: down-projection matrix
- $W^{UK} \in \mathbb{R}^{d_h n_h \times d_c}$: up-projection matrix for keys
- $W^{UV} \in \mathbb{R}^{d_h n_h \times d_c}$: up-projection matrix for values
- $W^{KR} \in \mathbb{R}^{d_h^R \times d}$
- $\mathrm{RoPE}\left(\cdot\right)$: RoPE matrices
- queryについても同様にlow-rank compressionにより以下のようにして得られる

$$
\begin{aligned}
\mathbf{c}_t^Q &= W^{DQ}\mathbf{h}_t, \\
[\mathbf{q}_{t,1}^C; \mathbf{q}_{t,2}^C; \ldots; \mathbf{q}_{t,n_h}^C] = \mathbf{q}_t^C &= W^{UQ}\mathbf{c}_t^Q, \\
[\mathbf{q}_{t,1}^R; \mathbf{q}_{t,2}^R; \ldots; \mathbf{q}_{t,n_h}^R] = \mathbf{q}_t^R &= \mathrm{RoPE}(W^{QR}\mathbf{c}_t^Q), \\
\mathbf{q}_{t,i} &= [\mathbf{q}_{t,i}^C; \mathbf{q}_{t,i}^R],
\end{aligned}
$$

- $c_t^Q \in \mathbb{R}^{d_c'}$: compressed latent vector for queries ($d_c' \ll d_h n_h$)
- d_c' : query compression dimension

- $W^{DQ} \in \mathbb{R}^{d'_c \times d}$: down-projection matrix
- $W^{UQ} \in \mathbb{R}^{d_h n_h \times d'_c}$: up-projection matrix
- $W^{QR} \in \mathbb{R}^{d^R_h n_h \times d'_c}$
- 最終的にattention配下のように計算される

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

$$\mathbf{u}_t = W^O[\mathbf{o}_{t,1}; \mathbf{o}_{t,2}; \ldots; \mathbf{o}_{t,n_h}],$$

DeepSeekMoE with Auxiliary-Loss-Free Load Balancing

- 全体図を見ればわかるように選択されるExpertsと必ず使われるExpertsがある
  - 前者をrouted experts
  - 後者をshared experts
- このときMoEは以下で定まる

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

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

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

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

- N_s : shared expertsの個数
- N_r : routed expertsの個数
- $\mathrm{FFN}_i^{(s)}$: i番目のshared experts
- $\mathrm{FFN}_i^{(r)}$: i番目のrouted experts

- k_r：routed expertsからtopkにより選択する個数
- これに対してauxiliary loss free load balancingを適用してgating scoreを以下で定める
  - https://arxiv.org/pdf/2408.15664

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

- 以下のcomplementary sequence-wise auxiliary lossを使用する

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

$$f_i = \frac{N_r}{K_r T} \sum_{t=1}^{T} \mathbb{1}\left(s_{i,t} \in \text{Topk}(\{s_{j,t} | 1 \leqslant j \leqslant N_r\}, K_r)\right),$$

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

$$P_i = \frac{1}{T} \sum_{t=1}^{T} s'_{i,t},$$

- αはハイパラで1は指示関数

Multi-Token Prediction (MTP)