Mohamed bin Zayed
University of
Artificial Intelligence

NLP Reading Group

October 15th 2025

“Modern LLMs - Part 1 (Architectures)”

Talk by: Nikolai Rozanov
(nikolai.rozanov@mbzuai.ac.ae)

Modern LLMs

Modern LLMs: Building Blocks

1. Architecture
    a. Positional Embeddings
    b. Attention
    c. Feed-forward
2. Data
    a. Pre-training
    b. Mid-training
    c. Post-training
3. Infrastructure
    a. Fault-tolerant Multi-node training
    b. Model-sharding
    c. Efficient GPU Code for Multi-node training & inference

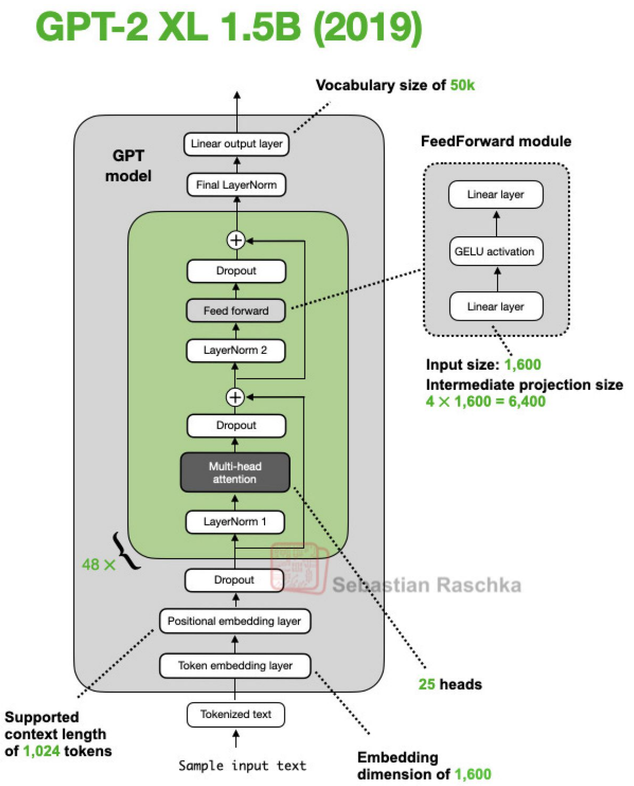
Focus Today!

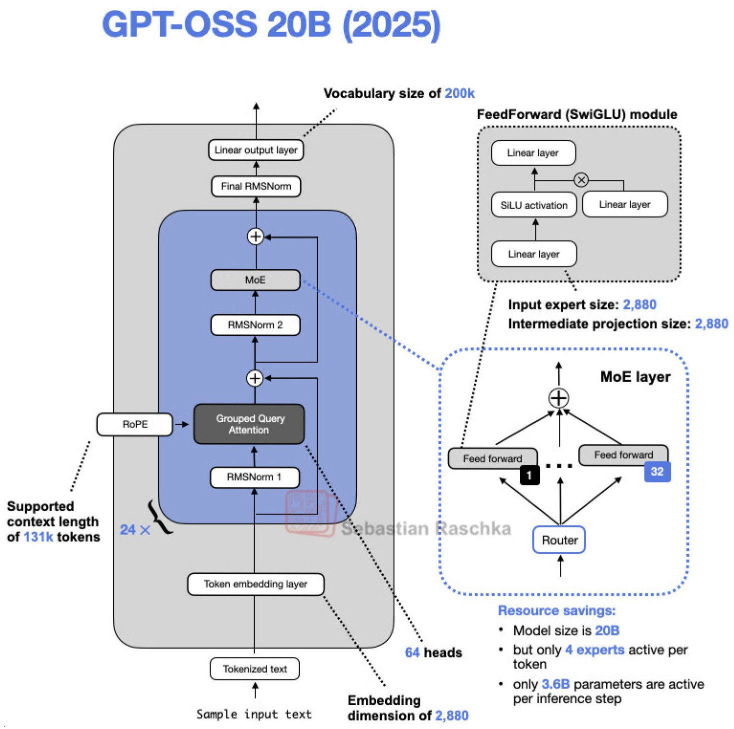
Modern LLMs: Architecture

1. Positional Embeddings
    a. Sinusoidal
    b. ROPE
    c. Yarn
2. Attention
    a. MHA
    b. MQA
    c. GQA
    d. MLA
    e. “Linear Attention” (State-Space Models, DeltaNet, Mamba)
3. “FFN”
    a. Classical FFN
    b. MoE
    c. Model-sharding
    d. Efficient GPU Code for Multi-node training & inference

Focus Today!

Modern LLMs: How to we go from GPT-2 -> GPT-OSS





https://magazine.sebastianraschka.com/p/from-gpt-2-to-gpt-oss-analyzing-the

Mohamed bin Zayed University of Artificial Intelligence

Attention Mechanisms

Multi-Head Attention (MHA)

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com
Noam Shazeer* Google Brain noam@google.com
Niki Parmar* Google Research nikip@google.com
Jakob Uszkoreit* Google Research usz@google.com
Llion Jones* Google Research llion@google.com
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu
Łukasz Kaiser* Google Brain lukaszkaiser@google.com
Illia Polosukhin* ‡ illia.polosukhin@gmail.com

"Attention is all you need"
https://arxiv.org/pdf/1706.03762

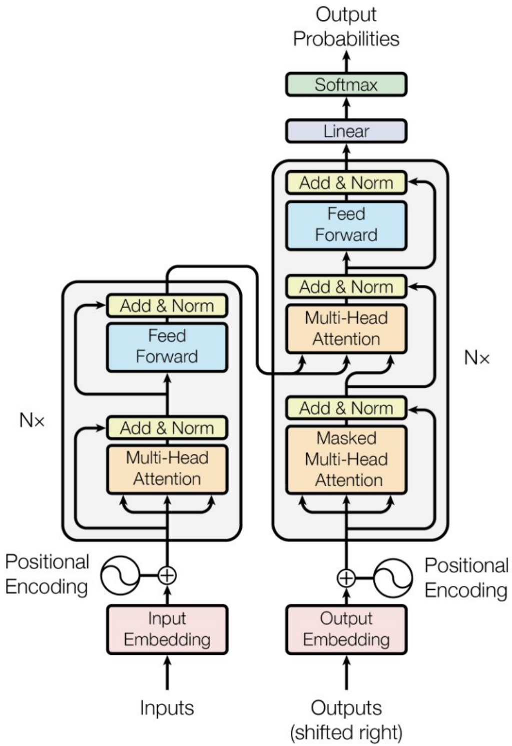
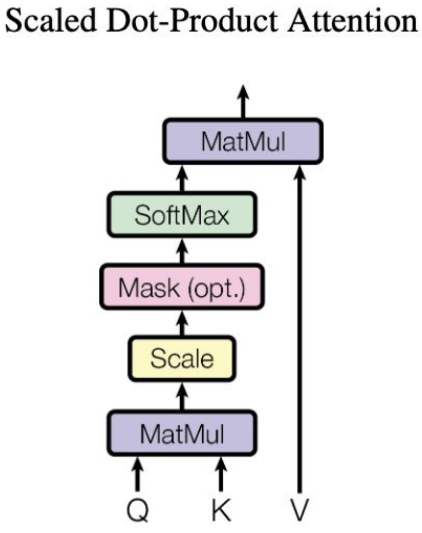
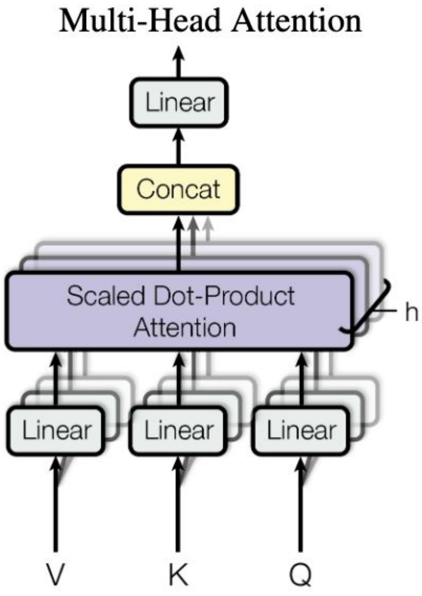


Figure 1: The Transformer - model architecture.

Multi-Head Attention (MHA)





"Attention is all you need"
https://arxiv.org/pdf/1706.03762

Multi-Head Attention (MHA)

Hi	How	are	Text
id=27	id=334	id=12	Tokens
emb_1	emb_2	emb_3	Embeddings

Q1 K1 V1   Q2 K2 V2   Q3 K3 V3

w11 = Q1 K1

w12 = Q1 K2

a11 $= \frac{e^{w_{11}}}{\sum_j e^{w1j}}$

h1 $= \sum_j a_{1j} v_j$

Multi-Head Attention (MHA)

	Hi	How	are	you
Text	Hi	How	are	you
Tokens	id=27	id=334	id=12	id=98
Embeddings	emb_1	emb_2	emb_3	emb_4
	Q1 K1 V1	Q2 K2 V2	Q3 K3 V3	Q4 K4 V4

w41 = Q4 K1

w42 = Q4 K2

$$a41 = \frac{e^{w_{41}}}{\sum_j w_{4j}}$$

$$h4 = \sum_j a_{4j} v_j$$

Mohamed bin Zayed
University of
Artificial Intelligence

Multi-Head Attention (MHA)

Hi	How	are	you	Text
id=27	id=334	id=12	id=98	Tokens
emb_1	emb_2	emb_3	emb_4	Embeddings
Q1 K1 V1	Q2 K2 V2	Q3 K3 V3	Q4 K4 V4	

$\rightarrow$

Computations needed to generate N2 tokens given N1 input tokens:
(N1+1)+(N1+2)+...+(N1+N2) = ~N^2 * Layers* *Heads*dim

Space Needed for KV-Cache:
2*N*Layers*Heads*dim

w41 = Q4 K1

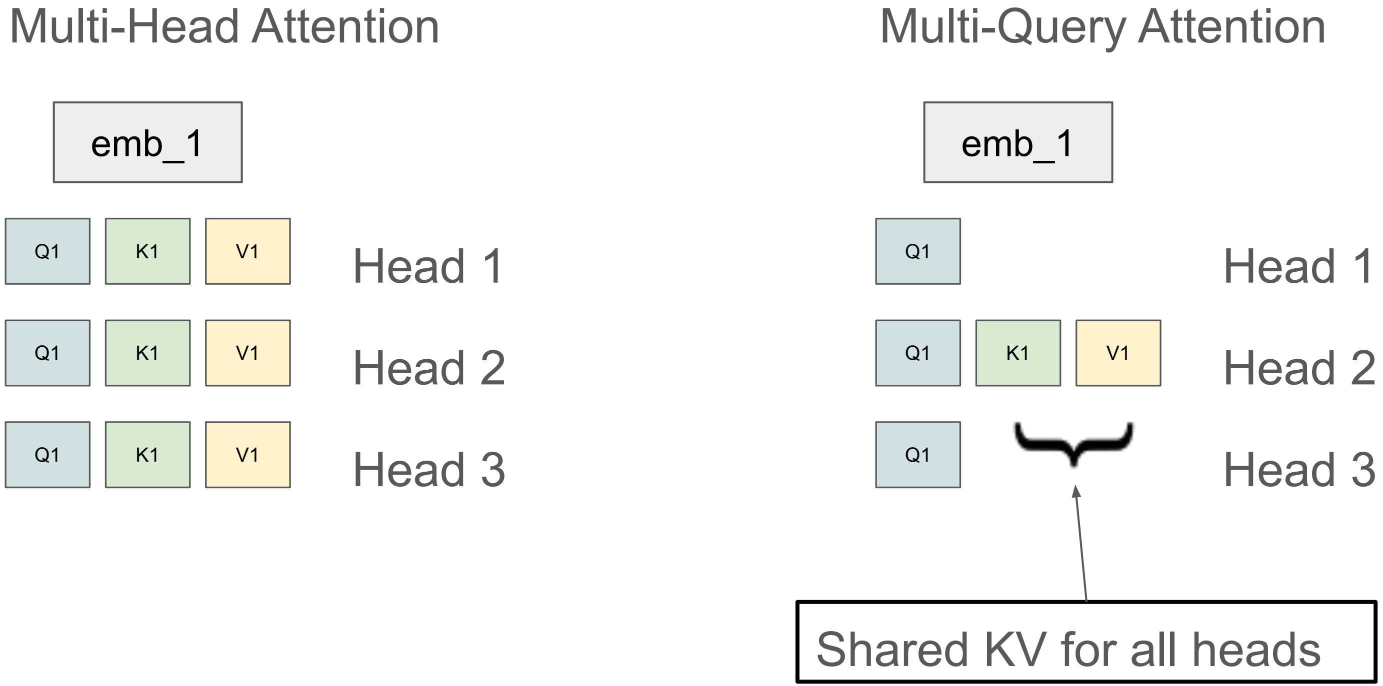
w42 = Q4 K2

$$a41 = \frac{e^{w_{41}}}{\sum_j w_{4j}}$$

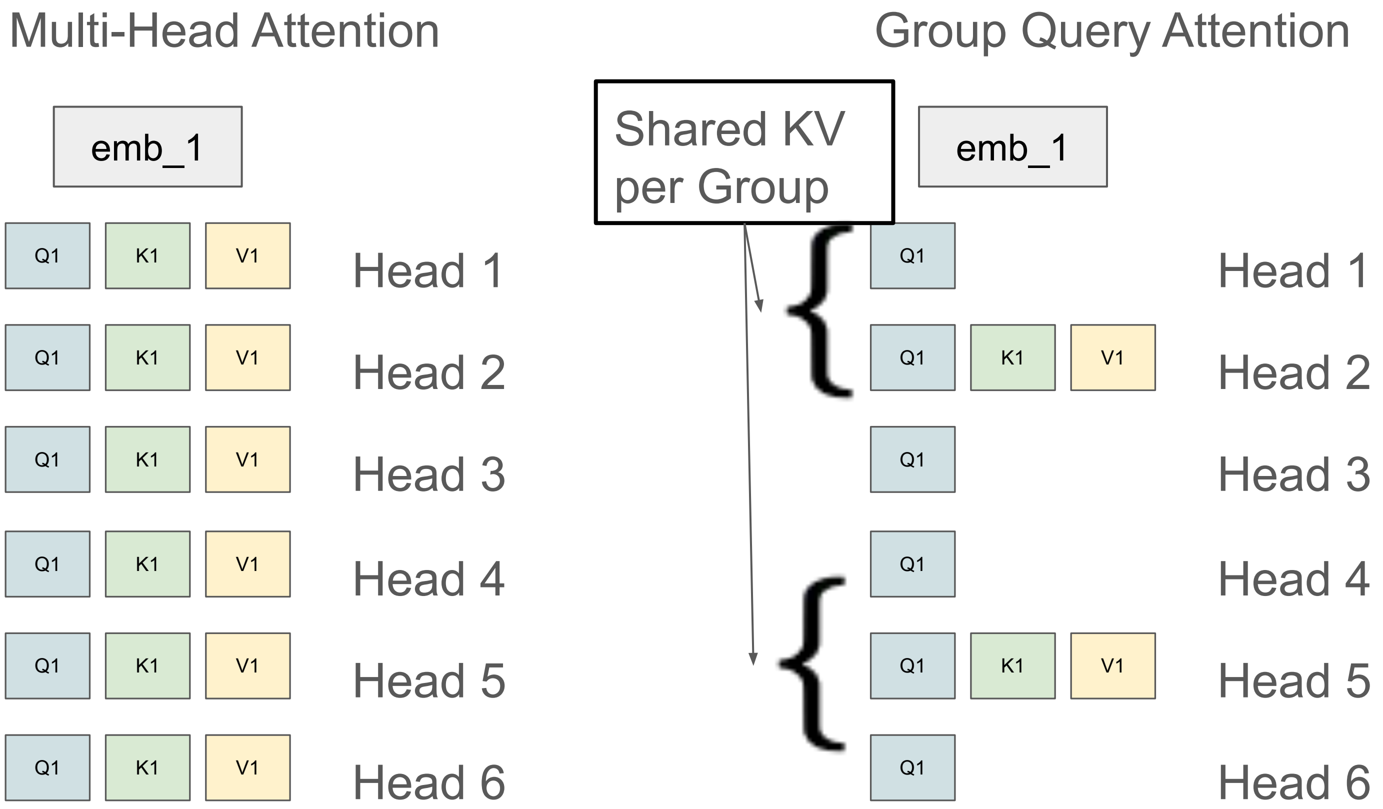
$$h4 = \sum_j a_{4j} v_j$$

Mohamed bin Zayed
University of
Artificial Intelligence

Multi-Query Attention (MQA)



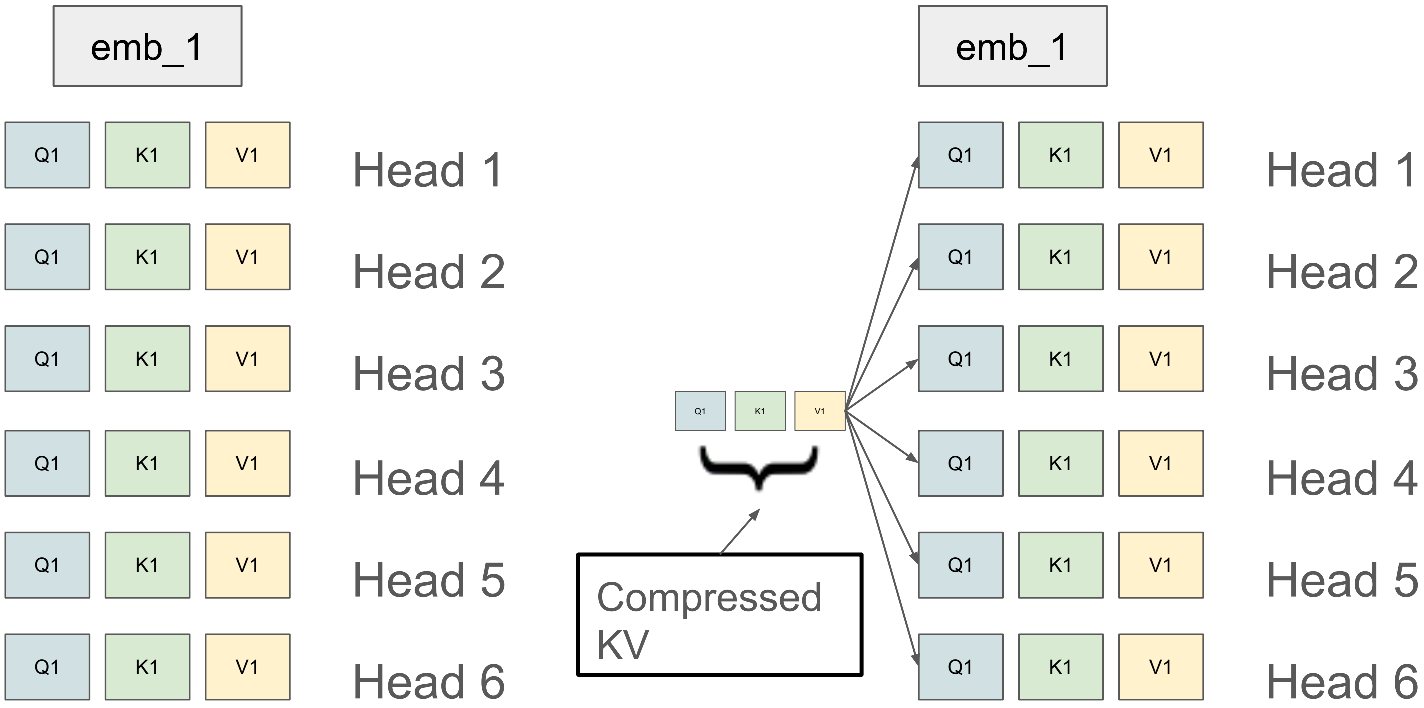
Group Query Attention (GQA)



Multi-Head Latent Attention (MLA)

Multi-Head Attention

Multi-Head Latent Attention

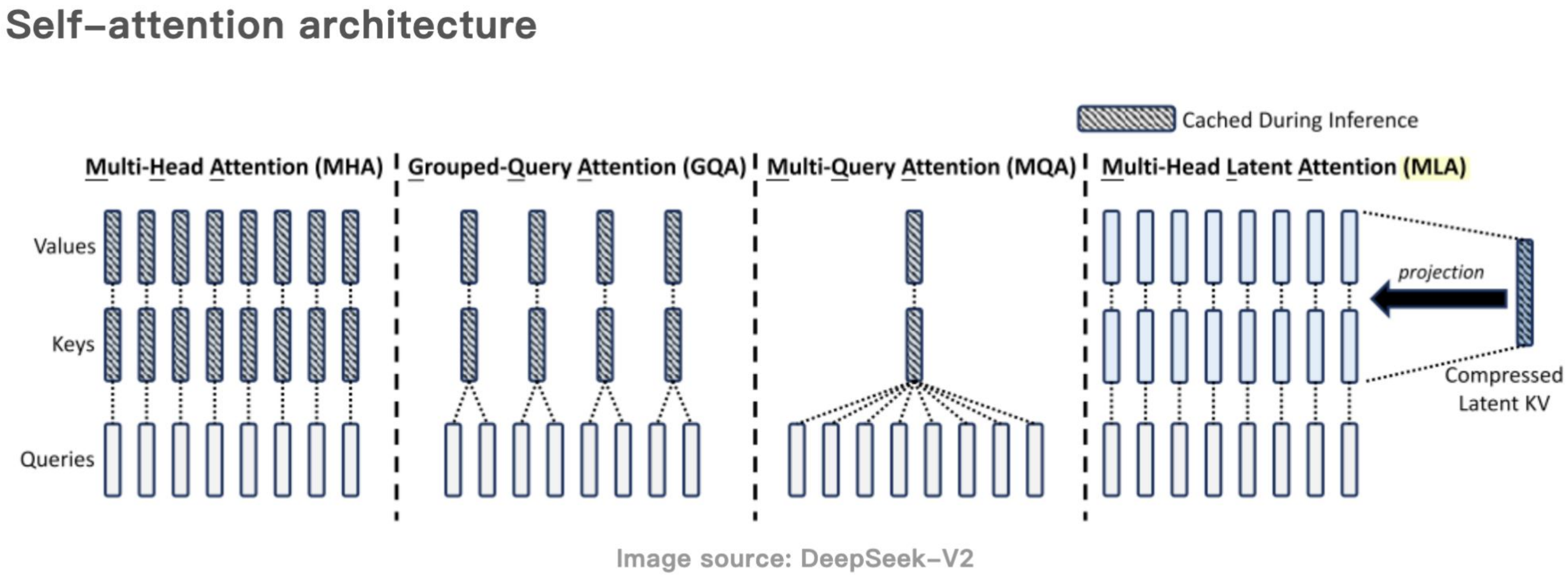


Mohamed bin Zayed University of Artificial Intelligence

Attention Comparison

Attention	Computation	Memory	Expressivity
Multi-Head Attention (MHA)	H x D x L x **N^2**	2 x H x D x L x N	High
Multi-Query Attention (MQA)	1 x D x L x **N^2**	2 x 1 x D x L x N	Low
Group-Query Attention (GQA)	G x D x L x **N^2**	2 x G x D x L x N	Medium
Multi-Head Latent Attention (MLA)	(H) x d x L x **N^2**	2 x d x L x N	Medium+

Attention Comparison



https://cyk1337.github.io/notes/2024/05/10/Memory-Efficient-Attention/

Alternative Attention Mechanisms:

1. Self-Attention:
    a. “Sliding Window” Attention
2. Linear Self-Attention (**State Space Models**)
    a. Mamba2 (Falcon-H1)
    b. DeltaNet (Qwen3-Next)
    c.

Model	Parameterization	Learnable parameters
Mamba (Gu & Dao, 2023)	$\mathbf{G}_t = \exp(-(\mathbf{1}^\top \boldsymbol{\alpha}_t) \odot \exp(\boldsymbol{A})), \quad \boldsymbol{\alpha}_t = \text{softplus}(\boldsymbol{x}_t \boldsymbol{W}_{\alpha_1} \boldsymbol{W}_{\alpha_2})$	$\boldsymbol{A} \in \mathbb{R}^{d_k \times d_v}, \quad \boldsymbol{W}_{\alpha_1} \in \mathbb{R}^{d \times \frac{d}{16}}, \quad \boldsymbol{W}_{\alpha_2} \in \mathbb{R}^{\frac{d}{16} \times d_v}$
Mamba-2 (Dao & Gu, 2024)	$\mathbf{G}_t = \gamma_t \mathbf{1}^\top \mathbf{1}, \quad \gamma_t = \exp(-\text{softplus}(\boldsymbol{x}_t \boldsymbol{W}_\gamma) \exp(a))$	$\boldsymbol{W}_\gamma \in \mathbb{R}^{d \times 1}, \quad a \in \mathbb{R}$
mLSTM (Beck et al., 2024; Peng et al., 2021)	$\mathbf{G}_t = \gamma_t \mathbf{1}^\top \mathbf{1}, \quad \gamma_t = \sigma(\boldsymbol{x}_t \boldsymbol{W}_\gamma)$	$\boldsymbol{W}_\gamma \in \mathbb{R}^{d \times 1}$
Gated Retention (Sun et al., 2024)	$\mathbf{G}_t = \gamma_t \mathbf{1}^\top \mathbf{1}, \quad \gamma_t = \sigma(\boldsymbol{x}_t \boldsymbol{W}_\gamma)^{\frac{1}{\tau}}$	$\boldsymbol{W}_\gamma \in \mathbb{R}^{d \times 1}$
DFW (Mao, 2022; Pramanik et al., 2023)	$\mathbf{G}_t = \boldsymbol{\alpha}_t^\top \boldsymbol{\beta}_t, \quad \boldsymbol{\alpha}_t = \sigma(\boldsymbol{x}_t \boldsymbol{W}_\alpha), \quad \boldsymbol{\beta}_t = \sigma(\boldsymbol{x}_t \boldsymbol{W}_\beta)$	$\boldsymbol{W}_\alpha \in \mathbb{R}^{d \times d_k}, \quad \boldsymbol{W}_\beta \in \mathbb{R}^{d \times d_v}$
GateLoop (Katsch, 2023)	$\mathbf{G}_t = \boldsymbol{\alpha}_t^\top \mathbf{1}, \quad \boldsymbol{\alpha}_t = \sigma(\boldsymbol{x}_t \boldsymbol{W}_{\alpha_1}) \exp(\boldsymbol{x}_t \boldsymbol{W}_{\alpha_2} \mathrm{i})$	$\boldsymbol{W}_{\alpha_1} \in \mathbb{R}^{d \times d_k}, \quad \boldsymbol{W}_{\alpha_2} \in \mathbb{R}^{d \times d_k}$
HGRN-2 (Qin et al., 2024b)	$\mathbf{G}_t = \boldsymbol{\alpha}_t^\top \mathbf{1}, \quad \boldsymbol{\alpha}_t = \boldsymbol{\gamma} + (1 - \boldsymbol{\gamma}) \sigma(\boldsymbol{x}_t \boldsymbol{W}_\alpha)$	$\boldsymbol{W}_\alpha \in \mathbb{R}^{d \times d_k}, \quad \boldsymbol{\gamma} \in (0,1)^{d_k}$
RWKV-6 (Peng et al., 2024)	$\mathbf{G}_t = \boldsymbol{\alpha}_t^\top \mathbf{1}, \quad \boldsymbol{\alpha}_t = \exp(-\exp(\boldsymbol{x}_t \boldsymbol{W}_\alpha))$	$\boldsymbol{W}_\alpha \in \mathbb{R}^{d \times d_k}$
Gated Linear Attention (GLA)	$\mathbf{G}_t = \boldsymbol{\alpha}_t^\top \mathbf{1}, \quad \boldsymbol{\alpha}_t = \sigma(\boldsymbol{x}_t \boldsymbol{W}_{\alpha_1} \boldsymbol{W}_{\alpha_2})^{\frac{1}{\tau}}$	$\boldsymbol{W}_{\alpha_1} \in \mathbb{R}^{d \times 16}, \quad \boldsymbol{W}_{\alpha_2} \in \mathbb{R}^{16 \times d_k}$

Table 1: Gated linear attention formulation of recent models, which vary in their parameterization of $\mathbf{G}_t$. The bias terms are omitted.

Source: Yang, Songlin, et al. “Gated linear attention transformers with hardware-efficient training.” arXiv preprint arXiv:2312.06635(2023).

Linear Attention Mechanisms: Core Idea ("Kernel Trick")

$Attention(Q, K, V) = softmax(QK^T)V$

$softmax(q^T k) = \frac{exp(q^T k)}{\sum_k exp(q^T k)}$

Key insight: We can approximate exponential similarity with kernel function ϕ:

$$exp(q^T k) \approx \phi(q)^T \phi(k)$$

For a single query-key-value triple:

$A(q, k, v) = \frac{\sum_{j=1}^{N} \phi(q)^T \phi(k_j) v_j}{\sum_{j=1}^{N} \phi(q)^T \phi(k_j)}$

Using matrix associativity:

$$(\phi(Q)\phi(K)^\top)V = \phi(Q)(\phi(K)^\top V) \quad (Assosiative\ property)$$

$$A(q, k, v) = \frac{\phi(q)^T \sum_{j=1}^{N} \phi(k_j) v_j^T}{\phi(q)^T \sum_{j=1}^{N} \phi(k_j)}$$

For full attention across all queries:

$$LinearAttention(Q, K, V) = \phi(Q)(\phi(K)^T V)$$

https://arxiv.org/pdf/2006.16236

https://medium.com/data-science/linearizing-attention-204d3b86cc1e

Feed Forward

Feed forward: Variants

Area of interest

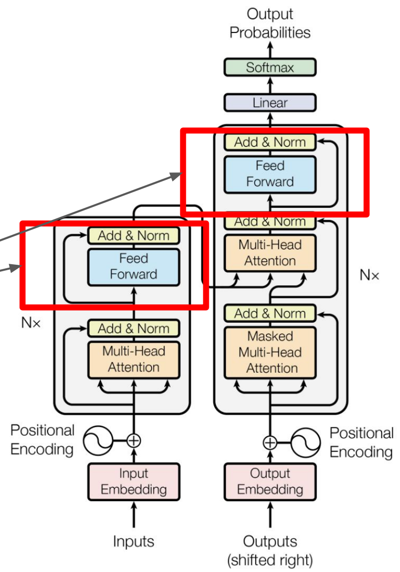


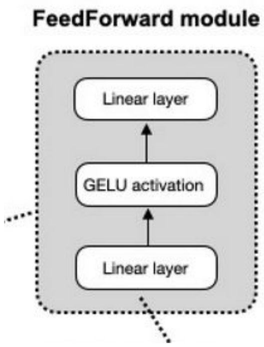
Figure 1: The Transformer - model architecture.

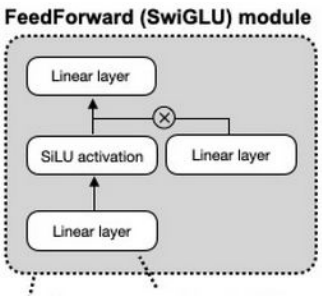
https://arxiv.org/pdf/1706.03762

Feed forward: Variants

1. Vanilla Feedforward
    a. GLU
    b. SwiGLU
2. Mixture of Experts (MoE)
    a. Shared Experts
    b. Sparse Experts
    c. Skip-Connection Experts (LongCat Chat)

Feed forward: Vanilla FeedForward to GLU FeedForward





https://magazine.sebastianraschka.com/p/the-big-llm-architecture-comparison

GLU Variants Improve Transformer

Noam Shazeer
Google
noam@google.com

February 14, 2020

$$
\begin{aligned}
\mathrm{FFN}_{\mathrm{GLU}}(x, W, V, W_2) &= (\sigma(xW) \otimes xV)W_2 \\
\mathrm{FFN}_{\mathrm{Bilinear}}(x, W, V, W_2) &= (xW \otimes xV)W_2 \\
\mathrm{FFN}_{\mathrm{ReGLU}}(x, W, V, W_2) &= (\max(0, xW) \otimes xV)W_2 \\
\mathrm{FFN}_{\mathrm{GEGLU}}(x, W, V, W_2) &= (\mathrm{GELU}(xW) \otimes xV)W_2 \\
\mathrm{FFN}_{\mathrm{SwiGLU}}(x, W, V, W_2) &= (\mathrm{Swish}_1(xW) \otimes xV)W_2
\end{aligned}
$$

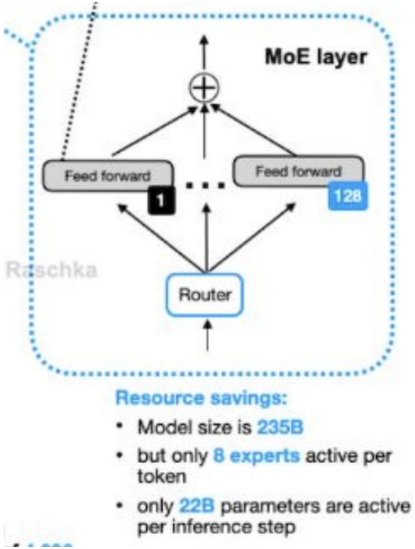
Training Steps	65,536	524,288
$\mathrm{FFN}_{\mathrm{ReLU}}$(*baseline*)	1.997 (0.005)	1.677
$\mathrm{FFN}_{\mathrm{GELU}}$	1.983 (0.005)	1.679
$\mathrm{FFN}_{\mathrm{Swish}}$	1.994 (0.003)	1.683
$\mathrm{FFN}_{\mathrm{GLU}}$	1.982 (0.006)	1.663
$\mathrm{FFN}_{\mathrm{Bilinear}}$	1.960 (0.005)	1.648
$\mathrm{FFN}_{\mathrm{GEGLU}}$	**1.942** (0.004)	**1.633**
$\mathrm{FFN}_{\mathrm{SwiGLU}}$	**1.944** (0.010)	**1.636**
$\mathrm{FFN}_{\mathrm{ReGLU}}$	1.953 (0.003)	1.645

https://arxiv.org/pdf/2002.05202

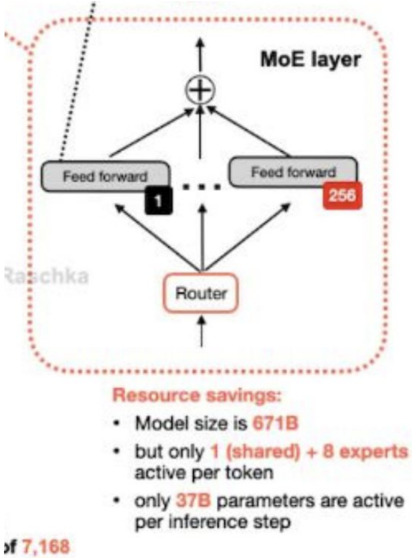
Mohamed bin Zayed University of Artificial Intelligence

Feed forward: Mixture of Experts

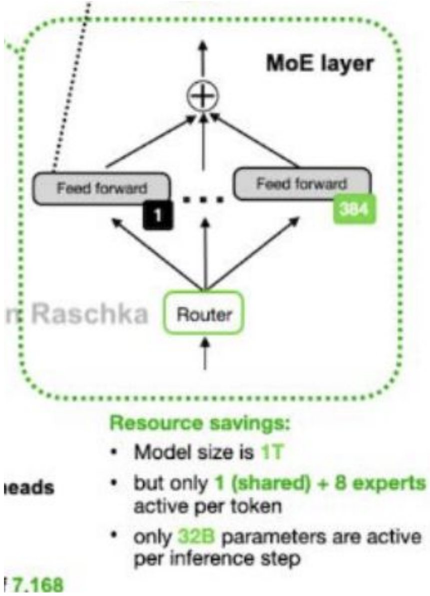
Qwen3



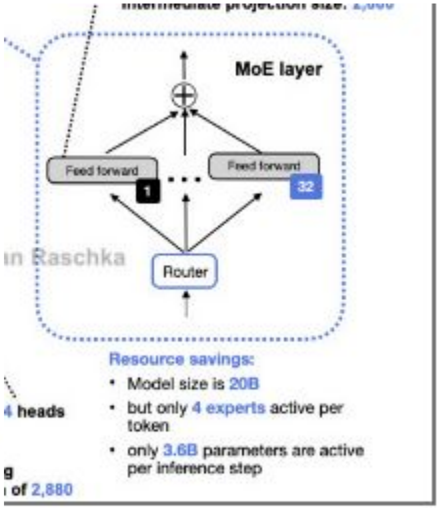
DeepSeek V3/R1



Kimi K2



GPT-OSS



https://magazine.sebastianraschka.com/p/the-big-llm-architect
ure-comparison

Feed forward: Beyond Mixture of Experts

"Zero-Computation Experts"


LongCat-Flash Technical Report

Meituan LongCat Team

longcat-team@meituan.com

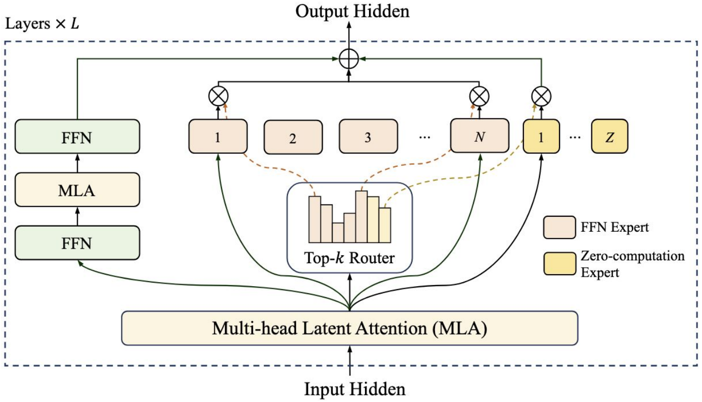
Additions:

1. Zero-Computation Experts
2. Additional MLA Block

$$\mathrm{MoE}(x_t) = \sum_{i=1}^{N+Z} g_i \, E_i(x_t),$$

$$g_i = \begin{cases} R(x_t)_i, & \text{if } R(x_t)_i \in \mathrm{TopK}\big(R(x_t)_i + b_i \mid 1 \le i \le N+Z, K\big), \\ 0, & \text{otherwise}, \end{cases}$$

$$E_i(x_t) = \begin{cases} \mathrm{FFN}_i(x_t), & \text{if } 1 \le i \le N, \\ x_t, & \text{if } N < i \le N+Z, \end{cases}$$



https://arxiv.org/pdf/2509.01322

Mohamed bin Zayed University of Artificial Intelligence

Main Architecture Summary

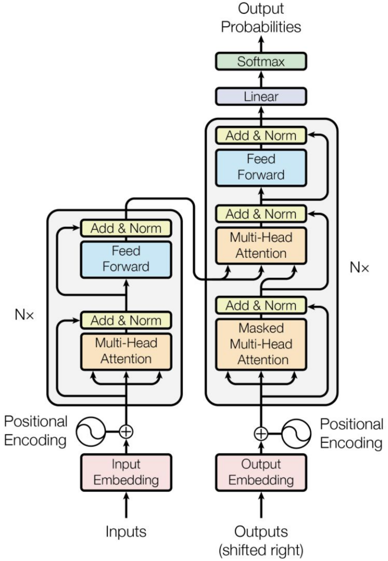
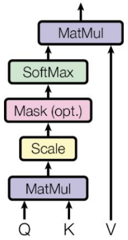


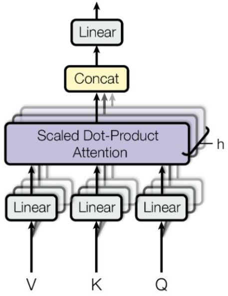
Figure 1: The Transformer - model architecture.

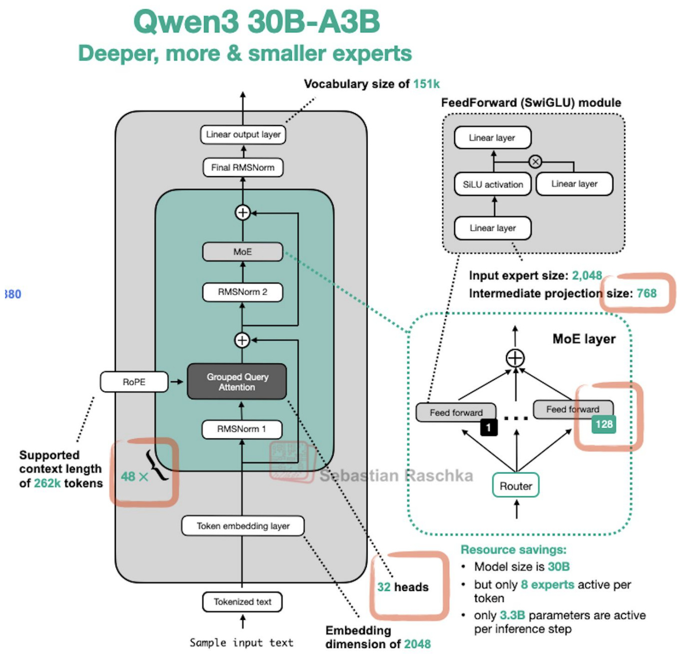
https://arxiv.org/pdf/1706.03762

Scaled Dot-Product Attention



Multi-Head Attention





https://magazine.sebastianraschka.com/p/the-big-llm-architecture-comparison

Mohamed bin Zayed University of Artificial Intelligence

Conclusion & Takeaways

Conclusions & Takeaways

1. Architectures keep evolving.

2. The key questions / challenges are:
    a. Memory scaling (to use all of the available memory efficiently)
    b. Computational scaling (to use compute efficiently)
    c. Both for Training and Inference

3. Model Architectures are becoming quite exotic:
    a. We are operating on the “Residual Stream”
    b. What other operators are useful?
    c. How to better scale speed and memory?

Main Architecture Trade-offs

1. Deep vs. Wide
    a. I.e. Many layers vs. large embedding sizes
2. Many Small Experts vs. Few Larger Experts
    a. Sparse Experts
3. Many Heads vs. Fewer Heads
4. Attention Mechanisms
    a. Linear vs. Classical
5. Other Trade-offs:
    a. Positional Embeddings
    b. Normalisation
    c. Activation Functions
6. Exotic Features:
    a. New Operators on Residual Stream
    b. New arrangement (beyond attention-feed_forward blocks)
    c. Maximising Memory / Compute use

References

Bibliography

- The Great LLM Comparison: https://magazine.sebastianraschka.com/p/the-big-llm-architecture-comparison
- Qwen2.5 https://arxiv.org/pdf/2412.15115
- DeepSeek 2.5 https://arxiv.org/pdf/2405.04434
- DeepSeek V3 https://arxiv.org/pdf/2412.19437
- Qwen 3 https://arxiv.org/pdf/2505.09388
- LongCat Chat https://arxiv.org/pdf/2509.01322
- Qwen3-Next https://qwen.ai/blog?id=3425e8f58e31e252f5c53dd56ec47363045a3f6b&from=research.research-list
- K2Think (Qwen2.5 Base) https://arxiv.org/pdf/2509.07604
- Attention Comparison https://cyk1337.github.io/notes/2024/05/10/Memory-Efficient-Attention/
- Falcon H1: https://arxiv.org/pdf/2507.22448

Mohamed bin Zayed University of Artificial Intelligence