IMPERIAL

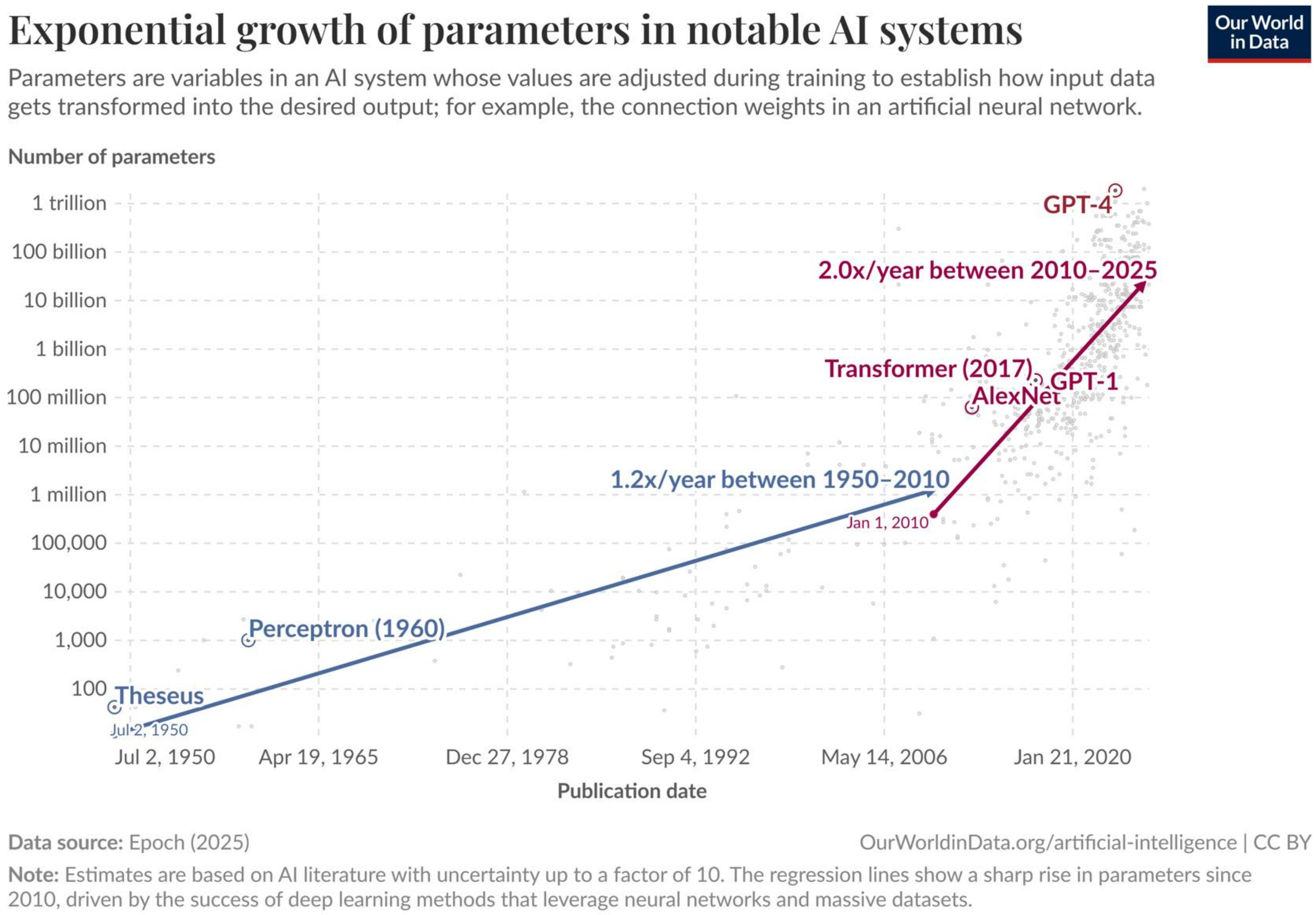
INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS
IJCNN2025
30 JUNE - 5 JULY 2025 | ROME, ITALY
INTERNATIONAL NEURAL NETWORK SOCIETY

TensorLLM: Tensorising Multi-Head Attention for Enhanced Reasoning and Compression in LLMs

Yuxuan Gu, Wuyang Zhou, Giorgos Iacovides, Danilo Mandic

Imperial College London

Why compression?



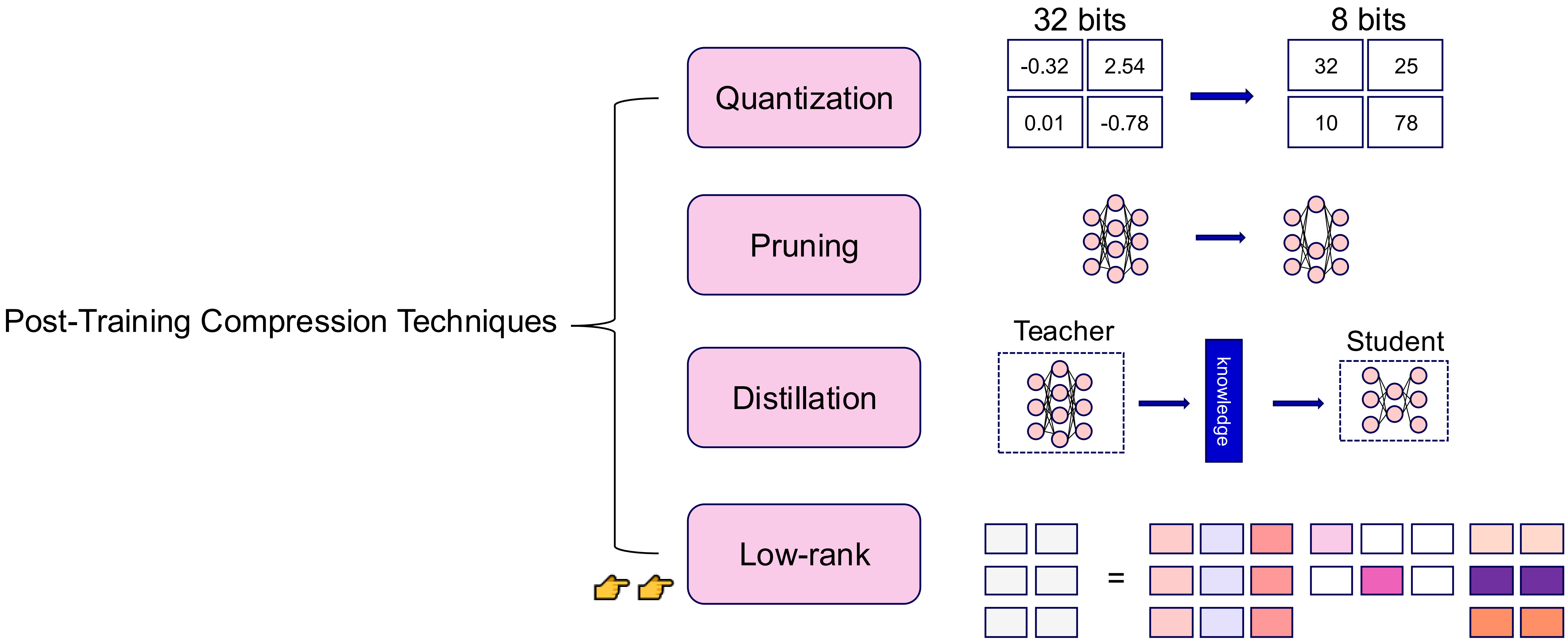
LLMs are too big for edge devices

Model	Size (FP16)	Typical Edge Storage	Deployable on Edge?
LLaMA-2 7B	14 GB	8–32 GB	Limited devices only
GPT-3	350 GB	-	Not deployable
PaLM 2 (Medium)	30 GB	8–32 GB	Too large
LLaMA-3 8B	16 GB	8–32 GB	Quantised only

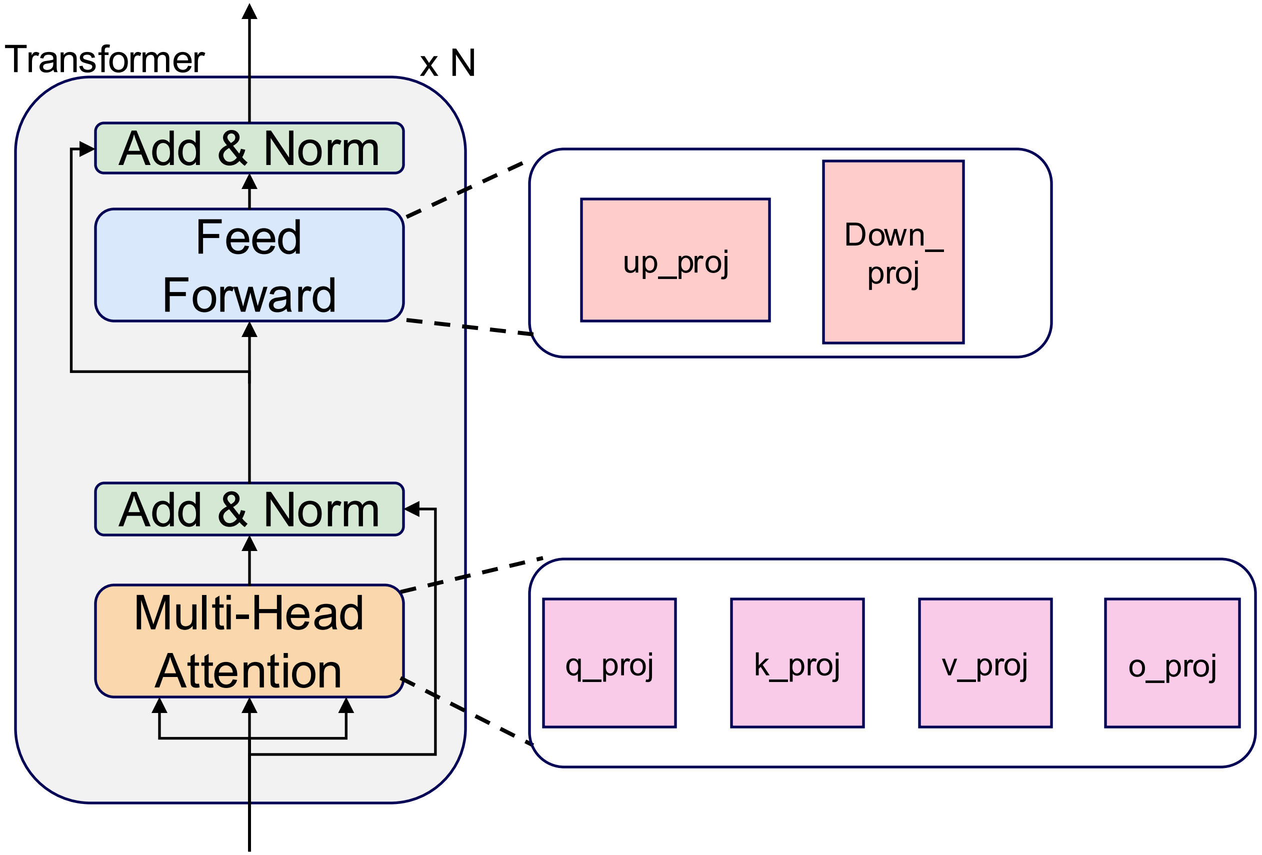
👉 Most edge devices don’t have enough memory or storage to host these models directly.

👉 Brute-force, data-driven engineering is becoming increasingly unsustainable.

How to compress?



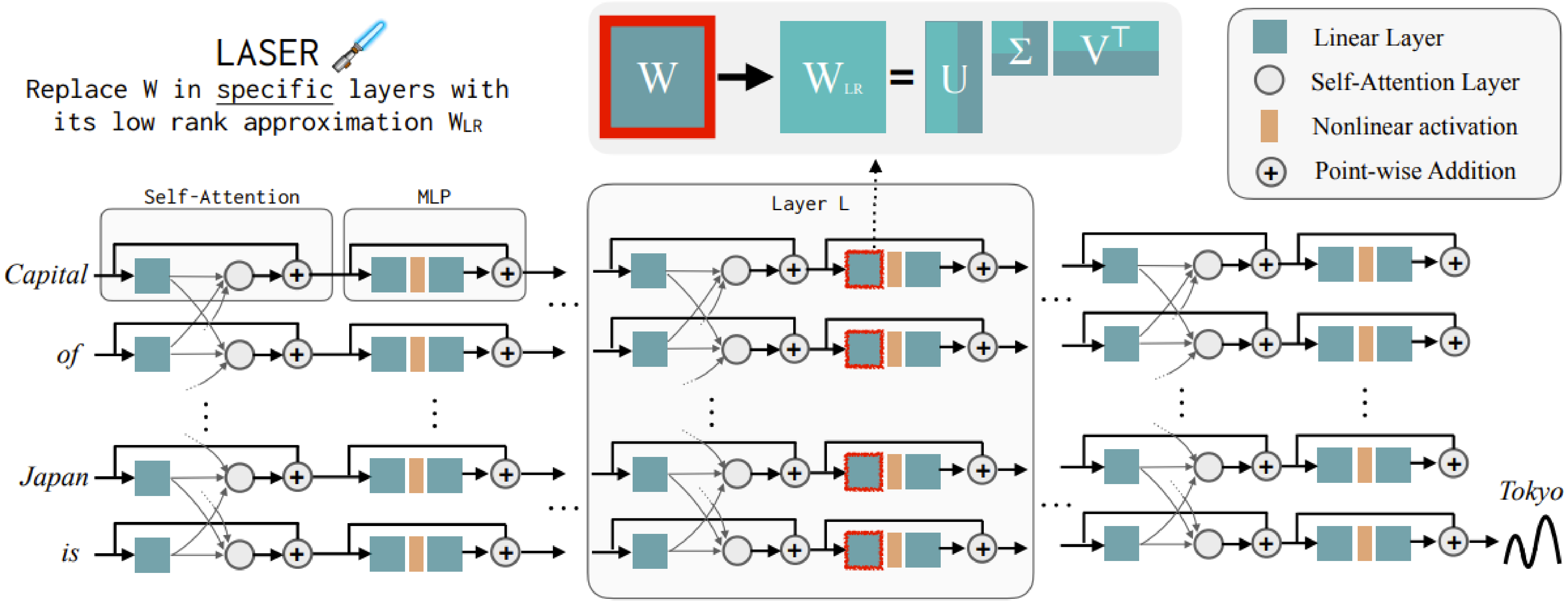
Over-parameterisation in LLMs



Model	Projection	Weight Shape
LLaMA-2 7B	q_proj	4096 × 4096
	k_proj	4096 × 4096
	v_proj	4096 × 4096
	o_proj	4096 × 4096
	up_proj	4096 × 11008
	down_proj	11008 × 4096

👉 Some of these weights might be redundant

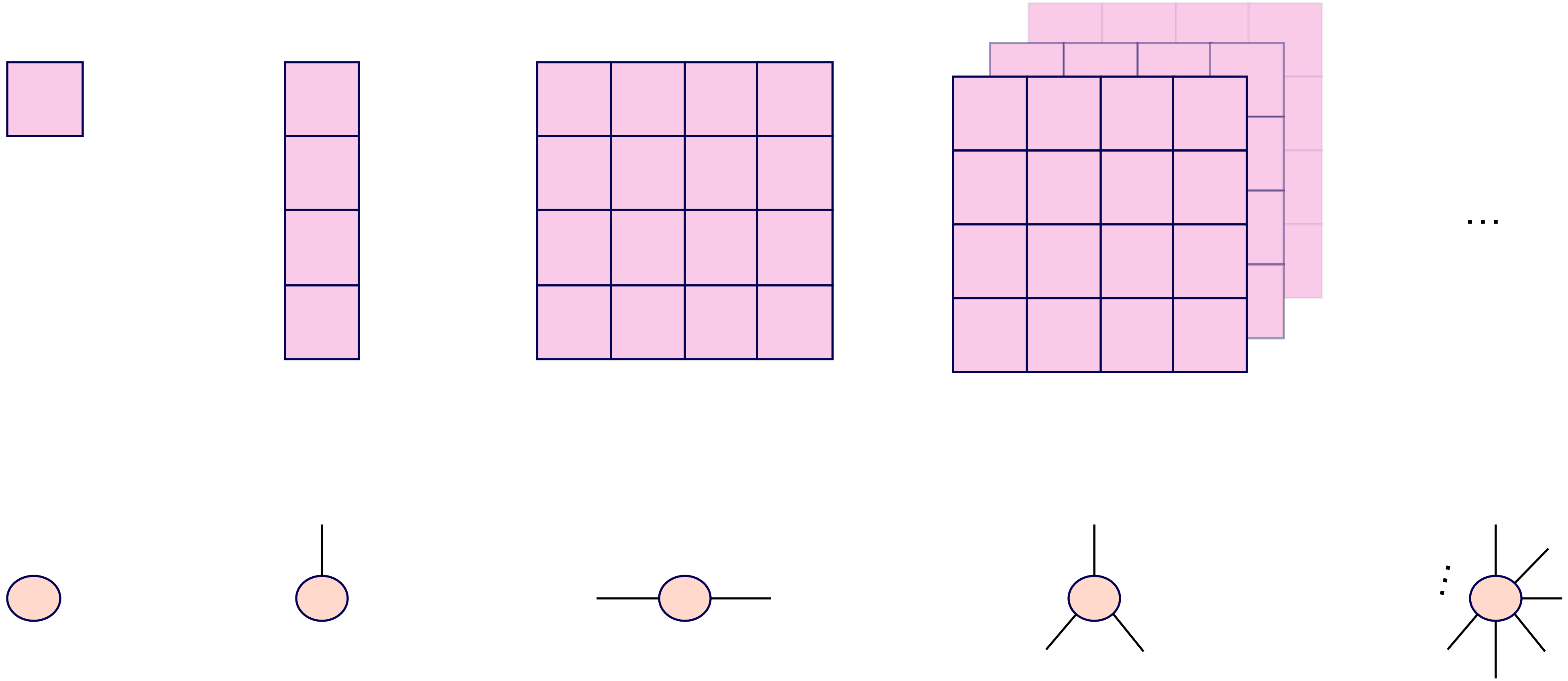
Related work: LASER
SVD for Post-Training Compression



👉👉 Can we explore inter-matrix redundancies as well?
Can we compress attention block as well?

Sharma, P., Ash, J.T. and Misra, D., The Truth is in There: Improving Reasoning in Language Models with Layer-Selective Rank Reduction. ICLR 2024

Tensor preliminaries



Scalar → 0D tensor

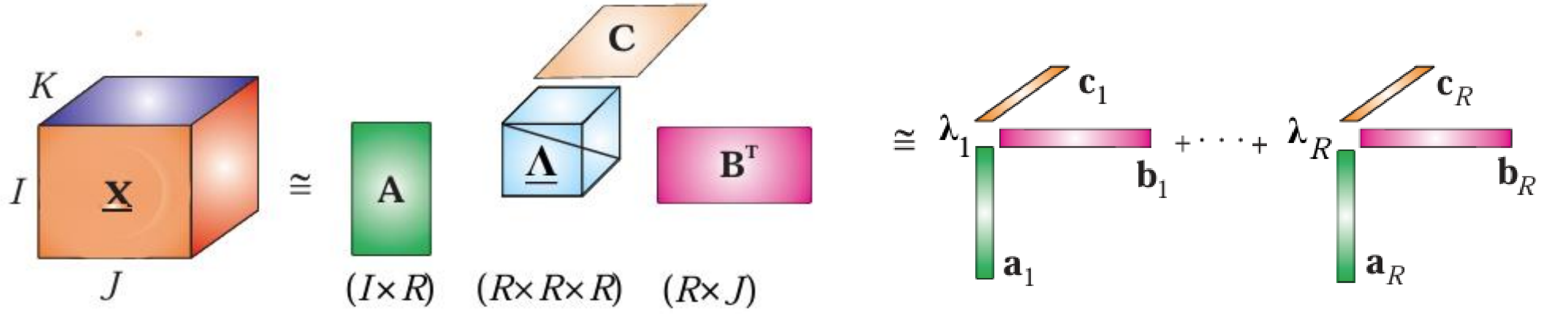
Vector → 1D tensor

Matrix → 2D tensor

3D tensor

ND tensor

Tensor preliminaries
Canonical Polyadic Decomposition (CPD)



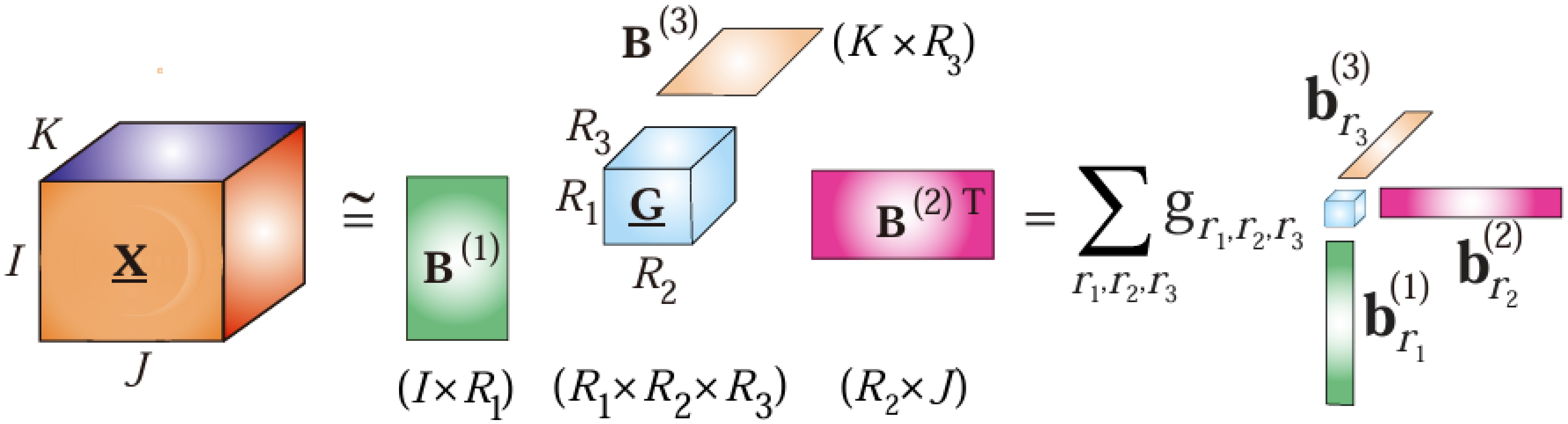
$$\mathcal{X} = \sum_{r=1}^{R} \lambda_r \, \boldsymbol{u}^{(1)} \circ \boldsymbol{u}^{(2)} \circ \cdots \circ \boldsymbol{u}^{(N)} = \mathcal{G} \times_1 \boldsymbol{U}^{(1)} \times_2 \boldsymbol{U}^{(2)} \times_N \boldsymbol{U}^{(N)}$$

Assume $\mathcal{X} \in \mathbb{R}^{I \times I \times \cdots \times I}$ is an order-N tensor. Each mode has rank R, where $R \ll I$.

Storage complexity: $\mathcal{O}(I^N) \rightarrow \mathcal{O}(R + NIR) \approx \mathcal{O}(NIR)$

Cichocki, A., Lee, N., Oseledets, I.V., Phan, A.H., Zhao, Q. and Mandic, D., 2016. Low-rank tensor networks for dimensionality reduction and large-scale optimization problems: Perspectives and challenges part 1. *arXiv preprint arXiv:1609.00893*.

Tensor preliminaries
Tucker Decomposition (TD)



$$\mathcal{X} = \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \cdots \sum_{r_N=1}^{R_N} \mathcal{G}_{[r_1,r_2,\cdots,r_N]} \boldsymbol{u}_{r_1}^{(1)} \circ \boldsymbol{u}_{r_2}^{(2)} \circ \cdots \circ \boldsymbol{u}_{r_N}^{(N)} = \mathcal{G} \times_1 \boldsymbol{U}^{(1)} \times_2 \boldsymbol{U}^{(2)} \times_N \boldsymbol{U}^{(N)}$$

Assume $\mathcal{X} \in \mathbb{R}^{I\times I\times\cdots\times I}$ is an order-N tensor. Each mode has rank R, where $R \ll I$.

Storage complexity: $\mathcal{O}(I^N) \rightarrow \mathcal{O}(R^N + NIR)$

Cichocki, A., Lee, N., Oseledets, I.V., Phan, A.H., Zhao, Q. and Mandic, D., 2016. Low-rank tensor networks for dimensionality reduction and large-scale optimization problems: Perspectives and challenges part 1. *arXiv preprint arXiv:1609.00893*.

TensorLLM

Design intuition:
- Attention heads within the same layer capture the same level of patterns
- Different attention heads within the same layer learn different specialised knowledge

Can we improve the reasoning capabilities of LLMs by enforcing a shared higher-dimensional subspace among the weights of multiple attention heads within a single transformer layer?

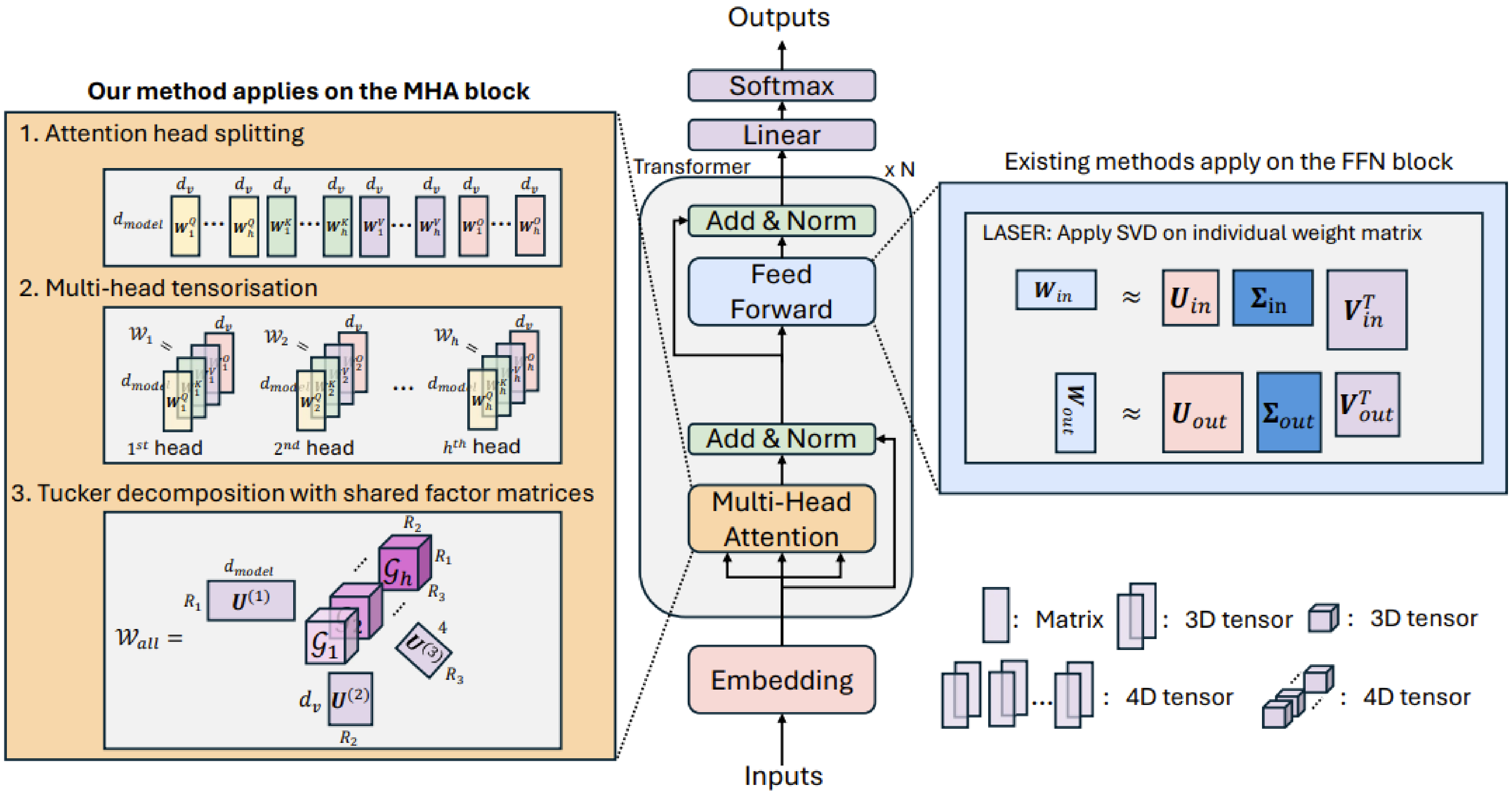
Gu, Y., Zhou, W., Iacovides, G. and Mandic, D., 2025. TensorLLM: Tensorising Multi-Head Attention for Enhanced Reasoning and Compression in LLMs. *IEEE International Conference on Neural Networks (IJCNN) 2025.*

TensorLLM

$$\mathbb{R}^{d_{model}\times h \cdot d_h}$$
$$\to \mathbb{R}^{d_{model}\times h \times d_h}$$

$$\mathcal{W}_{all} \in \mathbb{R}^{d_{model}\times d_h \times 4 \times h}$$

$$\mathcal{W}_{all} = \mathcal{G}_{all} \times_1 \boldsymbol{U}^{(1)}$$
$$\times_2 \boldsymbol{U}^{(2)} \times \boldsymbol{U}^{(3)} \times_4 \boldsymbol{I}$$



Gu, Y., Zhou, W., Iacovides, G. and Mandic, D., 2025. TensorLLM: Tensorising Multi-Head Attention for Enhanced Reasoning and Compression in LLMs. *IEEE International Conference on Neural Networks (IJCNN) 2025.*

TensorLLM
Original pretrained vs. Denoised Weights

Dataset		RoBERTa Original	RoBERTa Ours	GPT-J Original	GPT-J Ours	LLaMA2 Original	LLaMA2 Ours
HotPotQA	Acc	6.1	**7.33**	19.6	**20.15**	16.5	**18.44**
	Loss	10.99	**10.00**	**3.40**	4.49	**3.15**	9.80
	CR	-	1.12	-	247.30	-	3.54
FEVER	Acc	50.0	**50.45**	50.2	**58.94**	59.3	**66.75**
	Loss	2.5	**1.47**	1.24	**1.02**	1.02	**1.01**
	CR	-	3.74	-	14.69	-	3.54
Bios Profession	Acc	64.5	**72.57**	75.6	**81.18**	85.0	**86.61**
	Loss	**4.91**	6.64	4.64	**4.57**	**4.19**	4.54
	CR	-	8.78	-	74.68	-	3.54
BigBench-WikidataQA	Acc	28.0	**32.72**	51.8	**68.81**	59.5	**60.37**
	Loss	9.07	**8.72**	3.52	**2.63**	4.19	**2.38**
	CR	-	2.52	-	46.77	-	5.81

Gu, Y., Zhou, W., Iacovides, G. and Mandic, D., 2025. TensorLLM: Tensorising Multi-Head Attention for Enhanced Reasoning and Compression in LLMs. *IEEE International Conference on Neural Networks (IJCNN) 2025.*

TensorLLM
Combine with other methods

Dataset		RoBERTa Case 1	RoBERTa Case 2	RoBERTa Case 3 (Ours)	GPT-J Case 1	GPT-J Case 2	GPT-J Case 3 (Ours)	LLaMA2 Case 1	LLaMA2 Case 2	LLaMA2 Case 3 (Ours)
HotPotQA	Acc	6.7	5.24	**7.05**	19.5	19.62	**19.91**	17.2	18.88	**19.22**
	Loss	10.53	**8.60**	9.87	**3.39**	5.08	5.07	**2.97**	9.33	9.99
FEVER	Acc	52.3	53.6	**55.23**	56.2	55.59	**58.98**	64.5	65.13	**66.39**
	Loss	1.76	**1.18**	2.61	**1.27**	1.28	1.39	**0.91**	1.11	1.33
Bios Profession	Acc	72.5	71.14	**72.51**	82.1	81.28	**82.52**	86.7	86.07	**87.07**
	Loss	**6.44**	6.62	7.42	4.91	4.61	**4.52**	**4.05**	4.20	**4.05**
BigBench-WikidataQA	Acc	30.7	34.49	**37.40**	65.9	65.68	**68.20**	**62.0**	61.21	61.78
	Loss	**7.69**	8.25	7.86	2.86	2.89	**2.59**	**2.31**	2.35	2.34

Case 1: LASER was applied to 1 matrix in the FFN block
Case 2: LASER was applied to all matrices in the FFN and MHA blocks
Case 3: Our method was applied to the MHA block; LASER was applied to matrices in the FFN block

Gu, Y., Zhou, W., Iacovides, G. and Mandic, D., 2025. TensorLLM: Tensorising Multi-Head Attention for Enhanced Reasoning and Compression in LLMs. *IEEE International Conference on Neural Networks (IJCNN) 2025.*

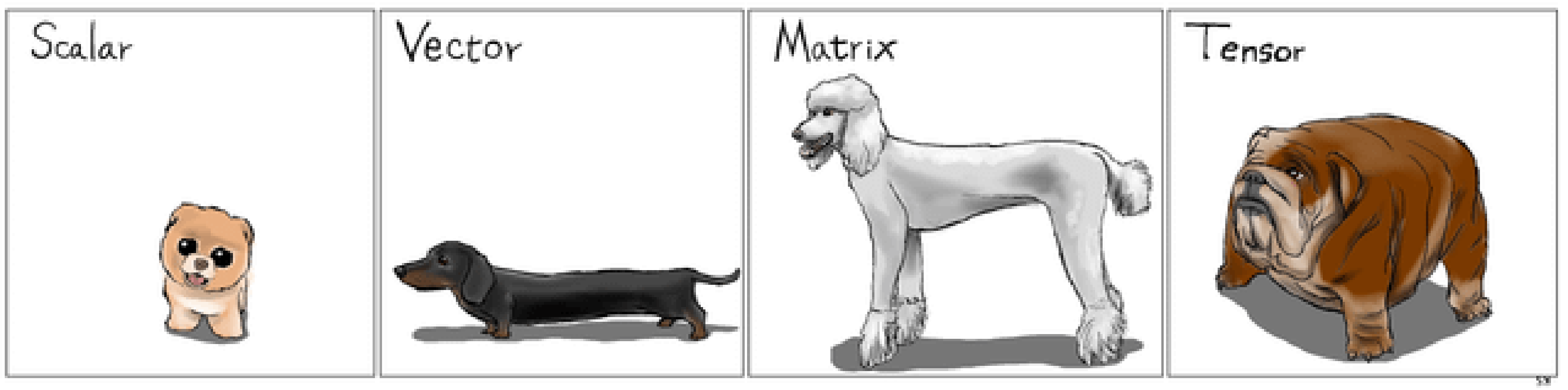
TensorLLM

Ablation study: whether to stack 4 matrices together

Dataset		GPT-J					
		Original	$\mathbf{W}^Q$	$\mathbf{W}^K$	$\mathbf{W}^V$	$\mathbf{W}^O$	Ours
HotPotQA	Acc	19.6	19.19	19.25	19.70	19.62	**20.15**
	Loss	**3.4**	4.45	4.45	4.43	4.44	4.49
FEVER	Acc	50.2	54.41	53.40	55.86	56.07	**58.94**
	Loss	1.24	1.22	1.22	1.23	1.15	**1.02**
Bios Profession	Acc	75.6	76.06	74.97	79.39	79.71	**81.18**
	Loss	4.64	4.54	4.59	4.46	**4.41**	4.57
BigBench-WikidataQA	Acc	51.8	49.72	51.01	48.82	48.87	**68.81**
	Loss	3.52	3.66	3.58	3.69	3.69	**2.63**

Gu, Y., Zhou, W., Iacovides, G. and Mandic, D., 2025. TensorLLM: Tensorising Multi-Head Attention for Enhanced Reasoning and Compression in LLMs. *IEEE International Conference on Neural Networks (IJCNN) 2025.*

IMPERIAL



Thank you !

Appendix

Tucker decomposition techniques

TensorLy

$$\mathcal{W}_{all} = \mathcal{G}_{all} \times_1 \boldsymbol{U}^{(1)} \times_2 \boldsymbol{U}^{(2)} \times \boldsymbol{U}^{(3)} \times_4 \boldsymbol{I}$$

$$\min_{\mathcal{G},\{U^{(i)}\}_{i=1}^{3}} \frac{1}{2}\left\|\mathcal{X} - \mathcal{G}_{all} \times_1 \boldsymbol{U}^{(1)} \times_2 \boldsymbol{U}^{(2)} \times \boldsymbol{U}^{(3)} \times_4 \boldsymbol{I}\right\|_F^2 \text{, subject to } U^{(n)^T} U^{(n)} = I_{R_N}\ \forall\, n \in [1, N]$$

Algorithm 1 Higher-Order Singular Value Decomposition (HOSVD)

Input: Tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, target ranks $R_1, R_2, \ldots, R_N$
Output: Core tensor $\mathcal{G}$ and factor matrices $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}$

- **for** $n = 1$ to N **do**
  - Unfold tensor $\mathcal{X}$ along mode n: $\mathbf{X}_{(n)} \in \mathbb{R}^{I_n \times (I_1 \cdots I_{n-1} I_{n+1} \cdots I_N)}$
  - Compute the top R_n left singular vectors of $\mathbf{X}_{(n)}$
  - Set $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times R_n} \leftarrow$ the resulting matrix
- **end for**
- Compute core tensor:

$$\mathcal{G} \leftarrow \mathcal{X} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \cdots \times_N \mathbf{U}^{(N)}$$

- **return** $\mathcal{G}, \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \cdots, \mathbf{U}^{(N)}$

Algorithm 2 Higher-Order Orthogonal Iteration (HOOI)

Input: Tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, target ranks $R_1, R_2, \ldots, R_N$
Output: Core tensor $\mathcal{G}$ and factor matrices $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}$

- Initialise $\mathbf{U}^{(n)} \in \mathbb{R}^{I_n \times R}$ for $n = 1, \ldots, N$ using HOSVD
- **repeat**
  - **for** $n = 1$ to N **do**
    - $\mathbf{Y} \leftarrow \mathcal{X} \times_1 \mathbf{U}^{(1)} \cdots \times_{n-1} \mathbf{U}^{(n-1)} \times_{n+1} \mathbf{U}^{(n+1)} \cdots \times_N \mathbf{U}^{(N)}$
    - $\mathbf{U}^{(n)} \leftarrow R_n$ leading left singular vectors of $\mathbf{Y}_{(n)}$
  - **end for**
- **until** variations of reconstruction error reaches a certain threshold
- $\mathcal{G} \leftarrow \mathcal{X} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \cdots \times_N \mathbf{U}^{(N)}$
- **return** $\mathcal{G}, \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}$

Refinement