KDD2025 August 3-7, 2025

上海交通大學 SHANGHAI JIAO TONG UNIVERSITY

RETHINK LAB SJTU-ReThinkLab 上海交通大学思维工场实验室

Reinvent the Operation not the Architecture: Quantum-inspired High-order Product for Compatible and Improved LLMs Training

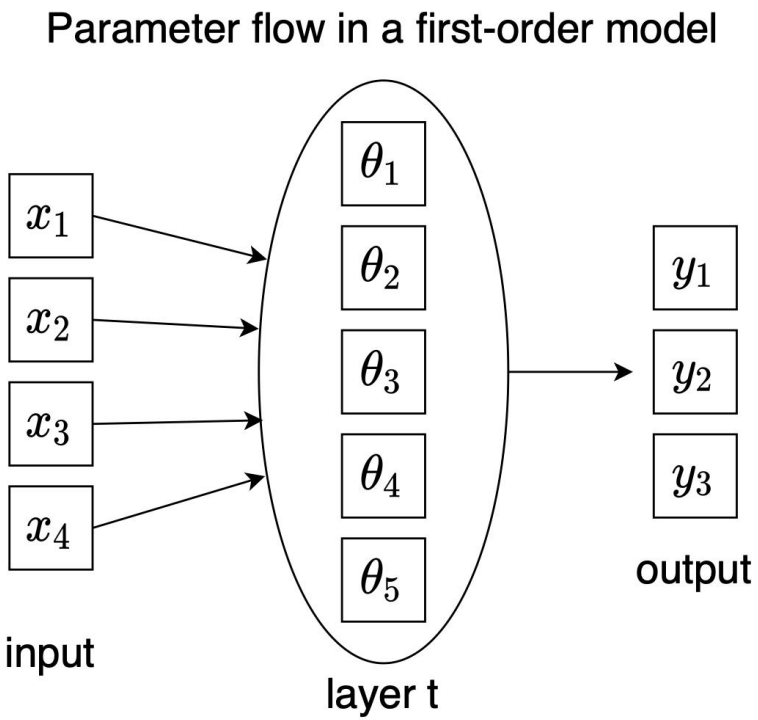
Hao Xiong*, Yebin Yang*, Huaijin Wu, Xiaoqiu Zhong, Yehui Tang, Zhuo Xia, Xiaoxing Wang, Junchi Yan.

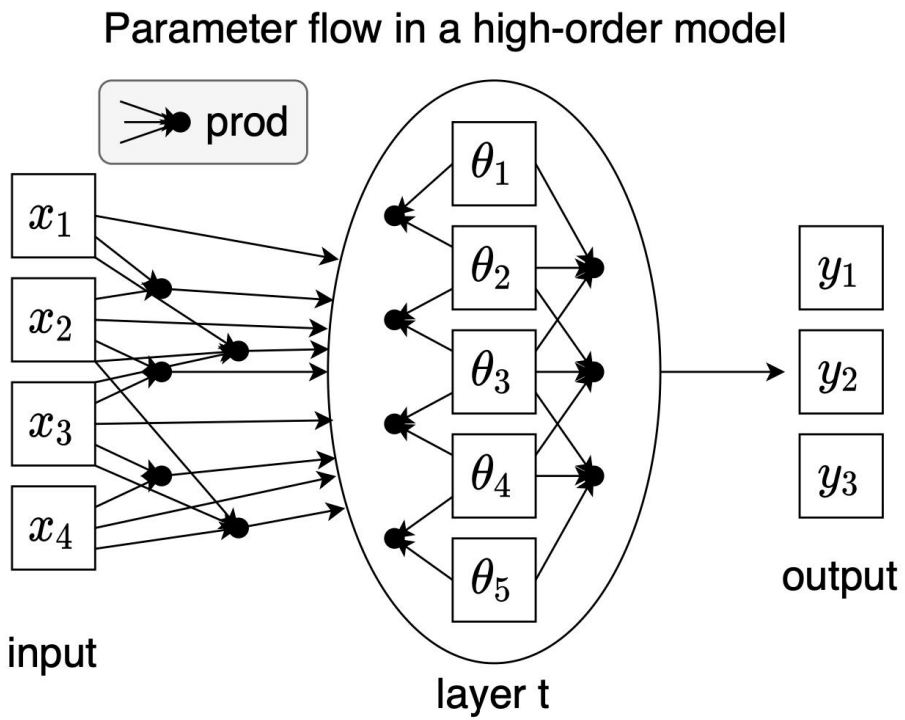
Highlighted features:

- A plug-in operator **HOLinear**, supporting both pre-training and finetuning
- Low complexity

Motivation

- Current NNs (e.g. Transformer)
  - Low-order
  - Parameter efficiency can be improved





Method (Part I)

- High–Order Embedding

$$\widehat{\mathbf{x}} = \bigotimes_{i=1}^{K} \begin{bmatrix} \mathbf{x}_i \\ 1 \end{bmatrix} = \mathcal{P}\left(\begin{bmatrix} 1 \\ \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_K \\ \mathbf{x}_1 \otimes \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_1 \otimes \mathbf{x}_2 \otimes \mathbf{x}_3 \\ \vdots \\ \mathbf{x}_1 \otimes \mathbf{x}_2 \otimes \cdots \otimes \mathbf{x}_K \end{bmatrix}\right)$$

$$= \mathcal{P}\left(\begin{bmatrix} \bigotimes_{\substack{\mathcal{S}\subseteq\{1,2,\ldots,K\} \\ |\mathcal{S}|\geq 2}} \mathbf{x}_i \\ \mathbf{x} \\ 1 \end{bmatrix}\right) \in \mathbb{R}^{\prod_{i=1}^{K}(d_i+1)},$$

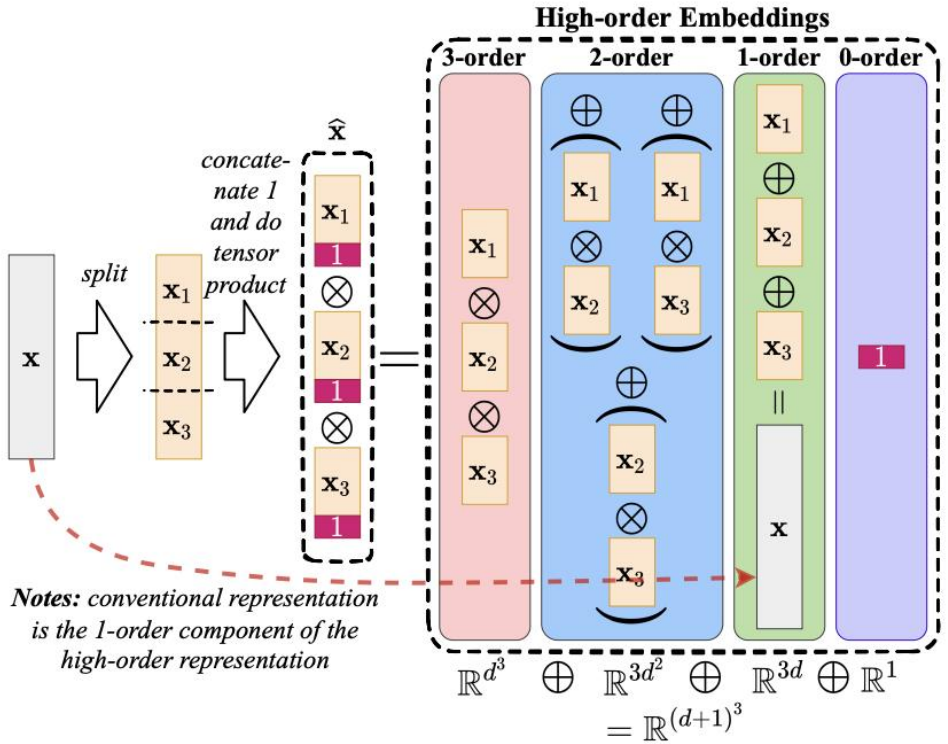


Figure 1: Constructing a high-orderly parameterized embedding for $K = 3$. The original low-order embedding $\mathbf{x}$ is the 1st-order component of the high-order embedding $\widehat{\mathbf{x}}$.

Method (Part II)

- HOLinear: High-Order Linear Mapping

$$\widehat{\mathbf{x}} \cdot \widehat{\mathbf{y}} = \prod_{i=1}^{K} \begin{bmatrix} \mathbf{x}_i \\ 1 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{y}_i \\ 1 \end{bmatrix} = \prod_{i=1}^{K} (\mathbf{x}_i \cdot \mathbf{y}_i + 1)$$

$$= 1 + \underbrace{\sum_{i=1}^{K} \mathbf{x}_i \cdot \mathbf{y}_i}_{=\mathbf{x}\cdot\mathbf{y}} + \sum_{\substack{\mathcal{S}\subseteq\{1,2,\ldots,K\} \\ |\mathcal{S}|\geq 2}} \prod_{i\in\mathcal{S}} \mathbf{x}_i \cdot \mathbf{y}_i,$$

$$\widehat{\mathbf{W}}^\top \widehat{\mathbf{x}} = \begin{bmatrix} \widehat{\mathbf{w}}_1 \cdot \widehat{\mathbf{x}} \\ \widehat{\mathbf{w}}_2 \cdot \widehat{\mathbf{x}} \\ \vdots \\ \widehat{\mathbf{w}}_M \cdot \widehat{\mathbf{x}} \end{bmatrix} \overset{①}{=} \begin{bmatrix} 1 + \sum_{\substack{\mathcal{S}\subseteq\{1,2,\ldots,K\} \\ |\mathcal{S}|\geq 1}} \prod_{i\in\mathcal{S}} \mathbf{w}_{1i} \cdot \mathbf{x}_i \\ 1 + \sum_{\substack{\mathcal{S}\subseteq\{1,2,\ldots,K\} \\ |\mathcal{S}|\geq 1}} \prod_{i\in\mathcal{S}} \mathbf{w}_{2i} \cdot \mathbf{x}_i \\ \vdots \\ 1 + \sum_{\substack{\mathcal{S}\subseteq\{1,2,\ldots,K\} \\ |\mathcal{S}|\geq 1}} \prod_{i\in\mathcal{S}} \mathbf{w}_{Mi} \cdot \mathbf{x}_i \end{bmatrix}$$

$$\overset{②}{=} 1 + \mathbf{W}^\top \mathbf{x} + \sum_{\substack{\mathcal{S}\subseteq\{1,2,\ldots,K\} \\ |\mathcal{S}|\geq 2}} \bigodot_{i\in\mathcal{S}} \mathbf{W}_i^\top \mathbf{x}_i \in \mathbb{R}^M,$$

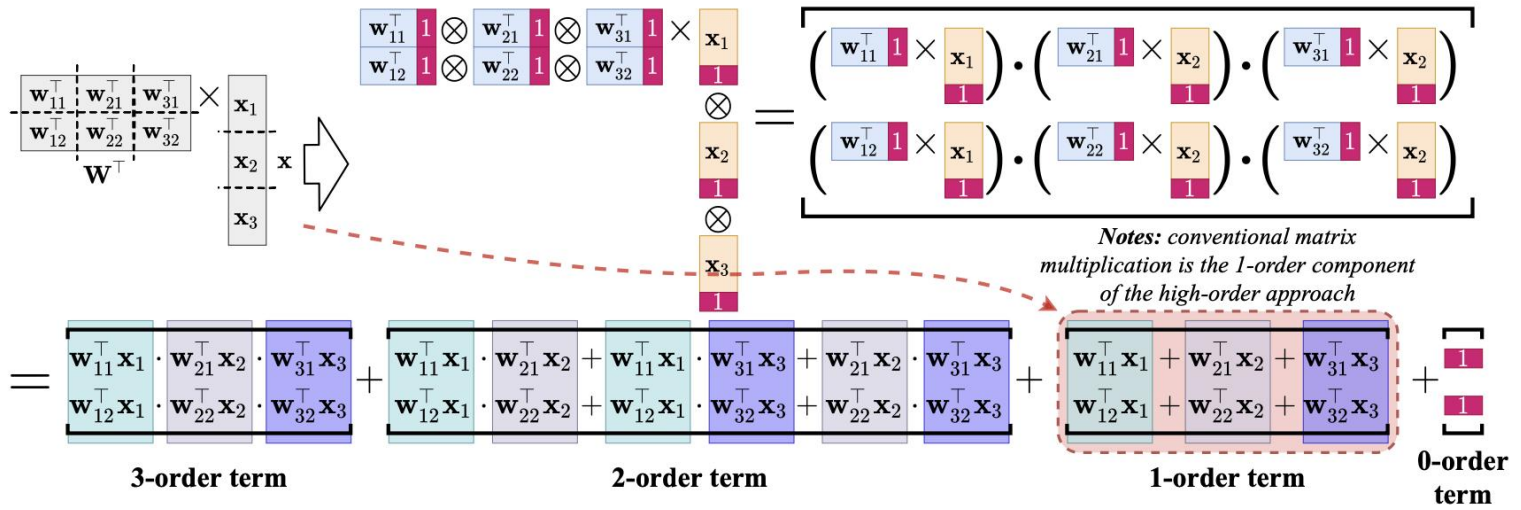
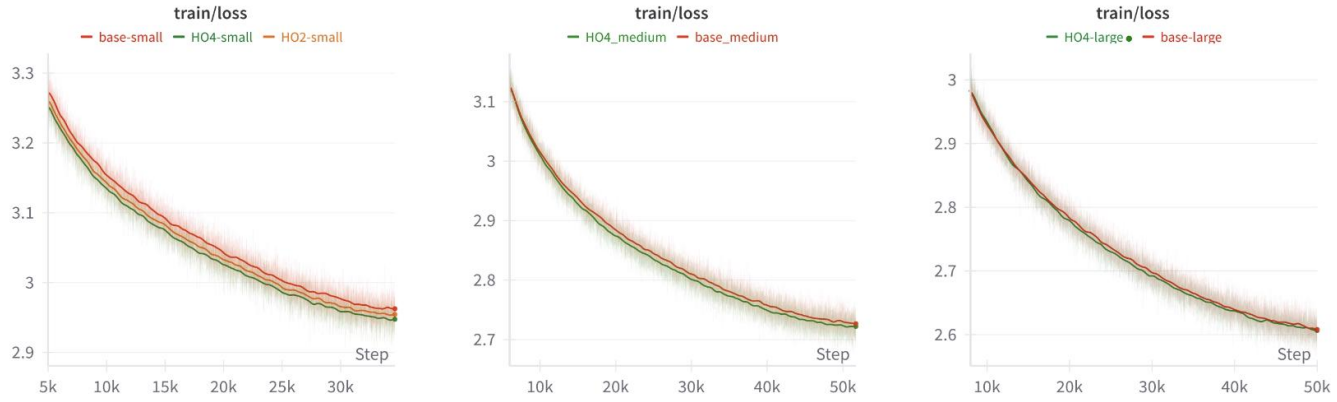


Figure 2: Toy example of order $K = 3$ for HOLinear which performs linear mapping for high-order vectors. The upper left panel shows the re-parameterization where the weight matrix M is transformed into the high-order $\widehat{\mathbf{W}}$, and the input vector x is transformed into $\widehat{\mathbf{x}}$. The upper right panel shows how the linear mapping result is mathematically derived without explicitly computing the tensor product. The bottom panel depicts the actual computation process, where each order's term is calculated before being summed together. Note that $\mathbf{w}_{mi}^\top \mathbf{x}_i$ is computed only once and reused across order terms for efficiency.

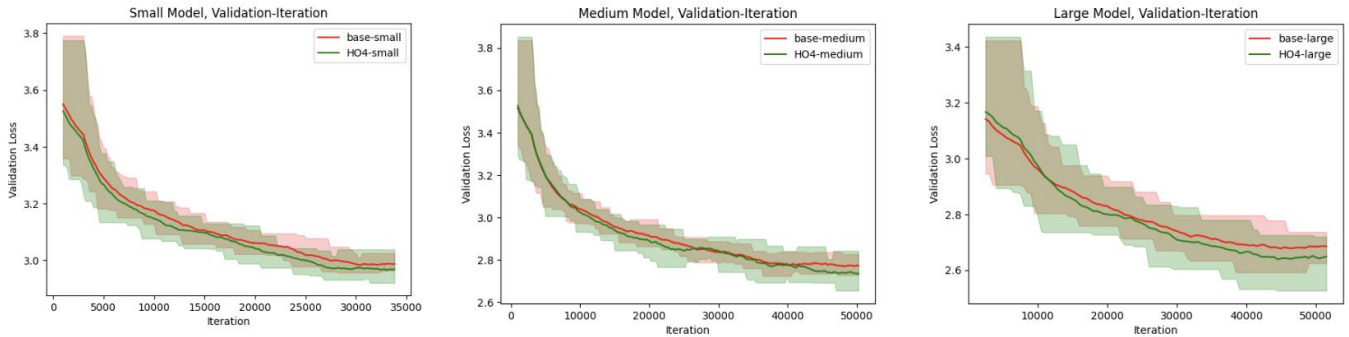
Experiments

- Pretraining



(a) small-size (124M) models (b) medium-size (353M) models (c) large-size (773M) models

Figure 4: Training loss comparison during pretraining of baseline models and HO-LLMs with different sizes on OpenWebText.



(a) small-size (124M) models (b) medium-size (353M) models (c) large-size (773M) models

Figure 5: Validation loss comparison in pretraining of baseline models and HO-LLMs by sizes on OpenWebText.

Experiments

- Pretraining

Table 3: Evaluation of models pre-trained on OpenWebText, across downstream tasks for small, medium, and large model sizes. The small-size model includes three choices: baseline, HO-LLM with HO order=2, and HO-LLM with HO order=4. Both medium-size and large-size models each include two choices: baseline and HO-LLM with HO order=4.

Model		Downstream Task Performance (Accuracy %)										
		arc_e.	arc_c.	hellaswag	mmlu	openbookqa	piqa	sciq	social_iqa	winogrande	wsc273	Avg.
Small (124M)	Baseline	42.72	19.62	30.52	**23.08**	26.20	61.81	**71.50**	36.95	49.96	53.11	41.55
	HO2	43.06	**20.73**	30.86	22.88	**27.60**	**62.46**	69.50	37.02	49.99	56.41	42.05
	HO4	**43.43**	20.56	**30.92**	22.83	27.00	61.70	71.10	**38.64**	**50.36**	**56.78**	**42.33**
Medium (353M)	Baseline	47.01	19.80	**31.10**	22.95	28.00	63.98	**74.60**	**38.89**	52.40	**58.24**	42.70
	HO4	**47.18**	**20.90**	31.08	**22.96**	**29.40**	**64.85**	74.20	38.38	**52.56**	57.51	**42.90**
Large (773M)	Baseline	**45.03**	**24.71**	**35.51**	22.96	29.40	66.05	77.30	39.66	**51.30**	59.97	45.19
	HO4	44.07	23.80	35.06	**22.98**	**29.60**	**66.86**	**77.80**	**39.87**	50.36	**62.27**	**45.27**

Experiments

- C2Q–SFT: Quantum–inspired HO Finetuning

Table 4: Accuracy (%) of Base LLaMA2-7B, Standard SFT (Std. SFT), and C2Q-SFT on downstream tasks. For HellaSwag, PIQA, WinoGrande, SciQ, MMLU, we evaluate models fine-tuned on Alpaca. For ARC-Easy, ARC-Challenge, and GSM8k, we evaluate models fine-tuned on GSM8k training set. Δ: relative improvement of C2Q-SFT over Standard SFT.

Task	Base	Std. SFT	C2Q-SFT	Δ
HellaSwag	57.55	60.81	**60.97**	+0.16
PIQA	**78.56**	78.51	**78.56**	+0.05
WinoGrande	**72.53**	70.24	70.32	+0.08
SciQ	96.80	96.60	**96.90**	+0.30
MMLU	45.01	46.38	**46.94**	+0.56
ARC-Easy	78.62	79.84	**80.18**	+0.34
ARC-Challenge	47.27	46.76	**47.35**	+0.59
GSM8k	10.84	38.89	**39.50**	+0.61
Average	60.90	64.75	**65.09**	+0.34

Experiments

- Combinatorial problems (TSP)

Table 5: Results on TSP-50 averaged over 1280 instances

Method	Avg. Length	Optimal Gap	Time
Concorde	5.688	0.000%	74ms
MatNet	5.721	0.587%	36ms
HO-MatNet	5.715	0.482%	61ms

results show the effectiveness and generalizability of HOLinear, demonstrating its potential beyond language modeling tasks.

- Thanks