A Simple Linear Patch Revives Layer-Pruned Large Language Models

Xinrui Chen[1], Haoli Bai[2], Tao Yuan[2], Ruikang Liu[1], Kang Zhao[2], Xianzhi Yu[2], Lu Hou[2], Tian Guan[1], Yonghong He[1], Chun Yuan[1]

[1]Tsinghua University, [2]Huawei Technologies

baihaoli@huawei.com, yuanc@sz.tsinghua.edu.cn

Code is available at https: //github.com/chenxinrui-tsinghua/LinearPatch

1. Background

- **Layer pruning** removes entire Transformer layers without requiring specialized kernels. Most methods suffer from severe performance degradation .
- We find the degradation mainly stems from **activation magnitude mismatch across the pruning interface**..

3. Method Overview — LinearPatch

Goal: Bridge activation magnitude mismatch at the pruning interface with a **lightweight plug-and-play patch**.

Core Idea: Fuse **Hadamard transformation** and **channel-wise scaling** into a single symmetric matrix:

$$X_{new}^{(\ell^*)} = X^{(\ell^*)}HDH^\top = X^{(\ell^*)}P$$

where $X^{(\ell^*)}$ is activation of layer ℓ^*, H is the Hadamard matrix, D is the diagonal scaling, and P is the **LinearPatch**.

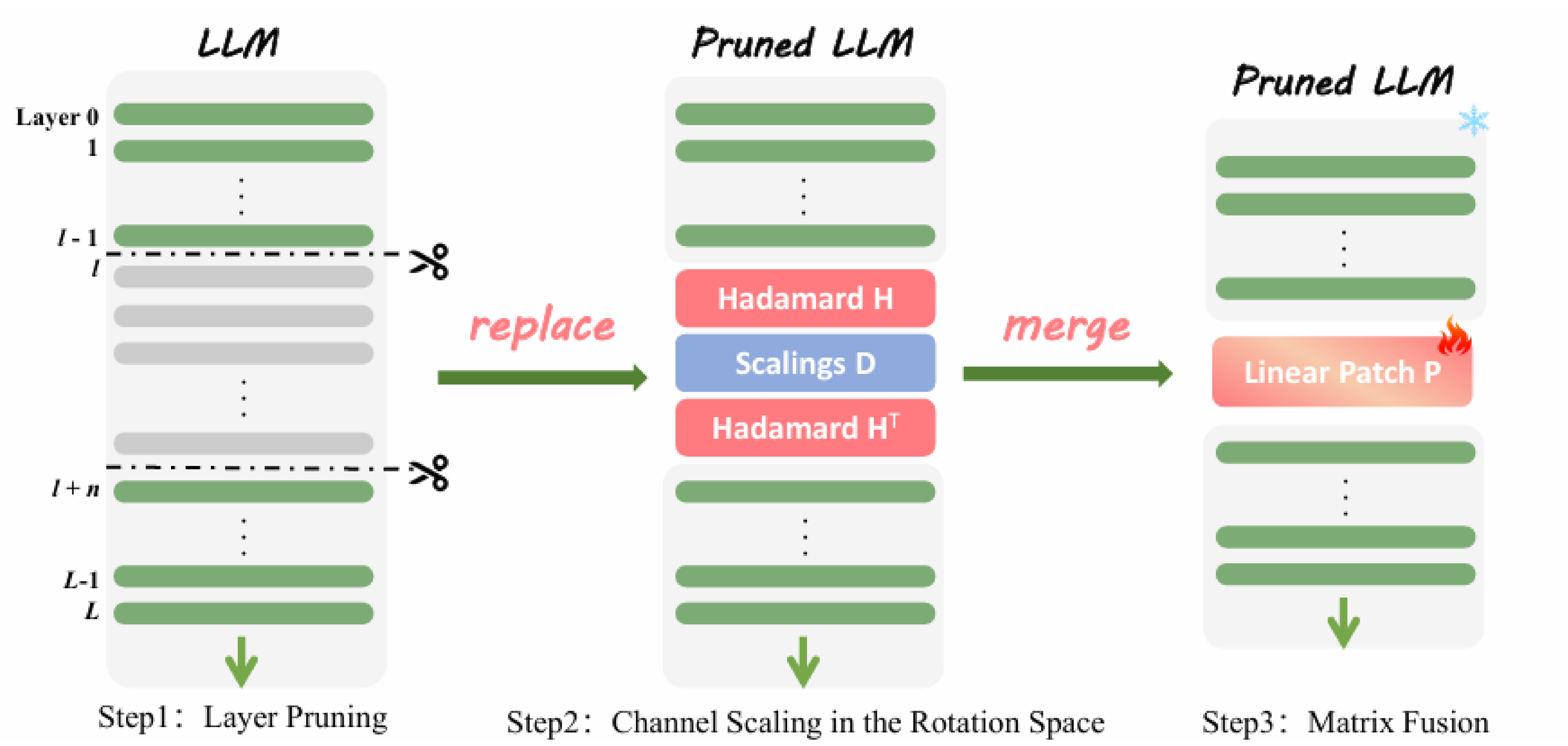

Figure 3: Overview of LINEARPATCH. First, layers are pruned using a specified metric. Next, channel-wise scalings are estimated in the Hadamard-transformed space. Finally, the scalings are fused with Hadamard transformations to form LINEARPATCH, which supports efficient fine-tuning.

2. Motivation

- **Observation:** The activations before and after the pruned layers exhibit drastically different scales →
Leads to distributional shift and instability in forward propagation.
- **Root causes:**
$\boxed{1}$ Channel-wise magnitude mismatch between layers.
$\boxed{2}$ Token-wise outlier activations (e.g., BOS, delimiter tokens).
- **Key question:**
How can we efficiently re-align activation magnitudes **without retraining the entire model**?

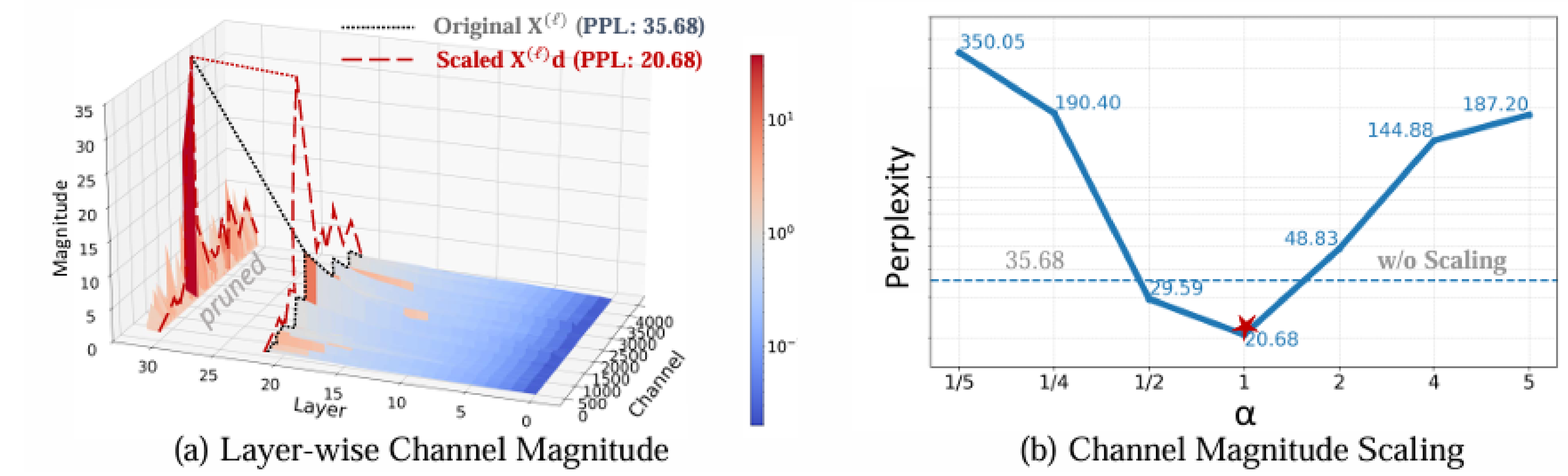

Figure 1: Visualization of layer-wise channel mismatch in pruned LLMs. Removing layers introduces magnitude mismatches, which we address using channel magnitude alignment.

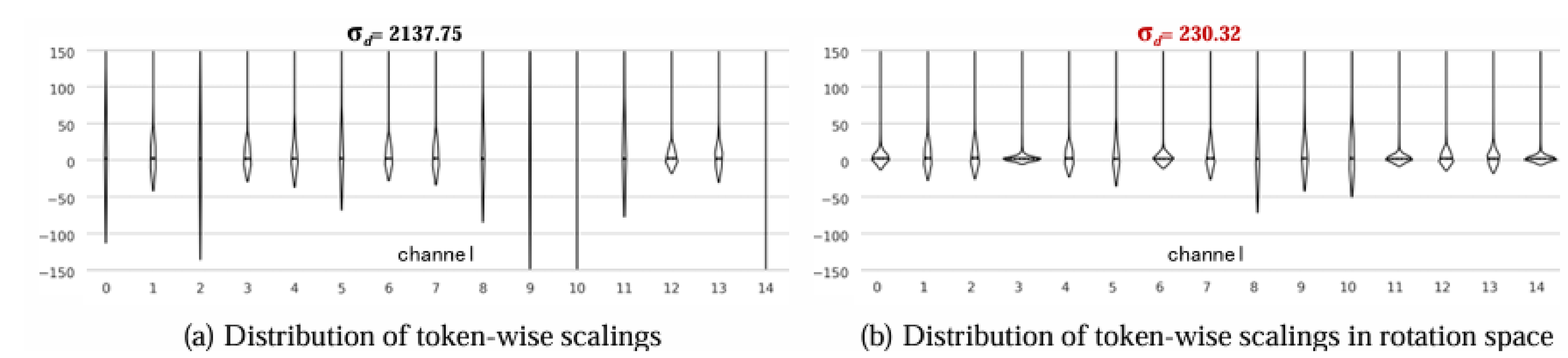

Figure 2: Violin plot of token-wise scaling mismatch in pruned LLMs. The violin width represents the estimated probability density of token-wise scalings. After applying the Hadamard transformation, the scaling distributions become more concentrated, indicating reduced variance across tokens.

4. Experiment

Table 1: Comparison on QA benchmark with training-free methods.

Model	L_p/L_t	Method	Ratio	ARC-c	ARC-e	BoolQ	HeSw	PIQA	WG	WSC	Race-h	CoPa	Avg.	RP
LLaMA-2-7B	0/32	Dense	-	46.25	74.58	77.74	75.97	79.11	68.98	80.59	39.62	87.00	69.98	100
	9/32	LLMPruner	26.99	31.91	52.90	62.42	54.41	71.33	53.20	65.57	28.52	79.00	55.47	78.14
	9/32	SLEB	27.03	31.91	52.31	46.09	58.28	69.59	58.25	69.23	32.25	79.00	55.21	78.41
	9/32	ShortGPT	27.03	32.76	48.61	62.17	56.17	64.36	64.33	71.06	32.25	77.00	56.52	80.29
	9/32	LLM-Streamline (None)	27.03	32.76	48.61	62.17	56.17	64.36	64.33	71.06	32.25	77.00	56.52	80.29
	9/32	LINEARPATCH$_{[S/L]}$	26.78	33.45	55.22	62.14	57.67	67.46	65.11	77.29	34.93	79.00	**59.14**	**84.08**
	7/32	LLMPruner	20.56	35.24	60.61	62.42	61.66	75.41	54.78	71.43	31.67	80.00	59.25	83.80
	7/32	SLEB	21.02	33.02	56.57	63.91	62.49	73.07	58.96	69.23	32.06	84.00	59.26	83.66
	7/32	ShortGPT	21.02	36.18	55.89	62.17	62.66	70.40	65.98	77.29	33.78	81.00	60.59	86.06
	7/32	LLM-Streamline (None)	21.02	36.18	55.89	62.17	62.66	70.40	65.98	77.29	33.78	81.00	60.59	86.06
	7/32	LINEARPATCH$_{[S/L]}$	20.78	37.63	61.24	62.14	63.49	70.46	65.90	79.49	36.46	85.00	**62.42**	**88.88**

Table 2: Comparison on QA benchmark with LLM-Streamline(FFN).

Model	L_p/L_t	Method	ARC-c	ARC-e	BoolQ	HeSw	PIQA	WG	WSC	Race-h	CoPa	Avg.	RP
LLaMa-2-7B	0/32	Dense	46.25	74.58	77.74	75.97	79.11	68.98	80.59	39.62	87.00	69.98	100
	7/32	LLM-Streamline + FT	38.23	60.48	70.18	63.75	69.86	67.48	80.95	37.51	79.00	63.05	90.00
	7/32	LINEARPATCH$_{[L]}$	37.63	61.24	62.14	63.49	70.46	65.90	79.49	36.46	85.00	62.42	88.88
	7/32	LINEARPATCH$_{[L]}$ + FT	38.23	64.35	65.32	69.33	73.23	67.40	83.88	38.37	87.00	**65.23**	**92.83**
LLaMA-3-8B	0/32	Dense	53.41	77.78	81.28	79.16	80.85	72.85	86.45	40.19	89.00	73.44	100
	5/32	LLM-Streamlines + FT	30.03	39.94	65.32	49.19	59.79	67.80	81.32	31.39	71.00	55.09	74.34
	5/32	LINEARPATCH$_{[L]}$	48.55	70.71	74.25	72.52	76.71	73.95	81.32	38.37	86.00	69.15	94.15
	5/32	LINEARPATCH$_{[L]}$ + FT	48.12	72.77	70.98	74.63	77.42	74.03	84.62	38.56	89.00	**70.01**	**95.16**

Table 3: Comparison on PPL benchmark with training-free methods.

Model	Method	WIKI-2	C4	PTB	PPL avg.
LLaMA-2-7B	Dense	5.47	6.97	22.51	11.65
	SLEB	9.14	11.21	38.45	19.60
	+LINEARPATCH	8.77	10.66	38.30	**19.24**
	Taylor+	18.45	20.99	62.18	33.87
	+LINEARPATCH	13.84	15.28	48.26	**25.79**
	ShortGPT	18.45	20.99	62.18	33.87
	+LINEARPATCH	13.22	14.58	45.97	**24.59**
	LLM-Streamline (None)	18.45	20.99	62.18	33.87
	+LINEARPATCH	13.22	14.58	45.97	**24.59**
LLaMA-3-8B	Dense	6.14	8.88	10.59	8.54
	SLEB	13.12	16.76	21.04	16.97
	+LINEARPATCH	11.97	15.74	19.55	**15.75**
	Taylor+	2287.86	1491.38	4741.90	2840.38
	+LINEARPATCH	208.88	235.63	264.97	**236.49**
	ShortGPT	57.76	50.13	67.39	58.43
	+LINEARPATCH	25.67	28.38	31.22	**28.42**
	LLM-Streamline (None)	2287.73	1491.37	4738.81	2839.30
	+LINEARPATCH	69.82	96.68	88.79	**85.10**

Table 4: Ablation study on ingredients.

	WIKI-2	C4	PTB	Avg.
Dense	5.47	6.97	22.51	11.65
Vanilla	35.68	36.10	96.52	56.10
+d	20.68	22.75	57.67	33.70
+P	18.60	19.28	53.00	30.29
+ FT	8.60	12.98	37.16	**19.58**