ReplaceMe: Training-Free Depth Pruning via Transformer Block Linearization

D. Shopkhoev[12], A. Ali[12], M. Zhussip[1], V. Malykh[123], S. Lefkimmiatis[1], N. Komodakis[456], S. Zagoruyko[7]
[1]MTS AI, [2]ITMO University, [3]IITU, [4]University of Crete, [5]IACM-Forth, [6]Athena RC, [7]Polynome

Introduction

LLMs are powerful but computationally expensive—limiting their real-world use due to **high latency**, **energy consumption**, and **hardware demands**.

Key Challenge:
- Depth pruning can shrink models by removing entire transformer blocks, but existing methods require **costly retraining** ("healing") to recover performance—defeating the purpose of efficient compression.
- Prior works (UIDL, LLM-Streamline) show that LLM have redundant blocks that can be removed, but it either leads to **major performance degradation** or architectural **modifications with fine-tuning**.

We propose **ReplaceMe**—a training-free depth-pruning method that replaces pruned blocks with a single linear transformation, estimated from a tiny calibration set. Up to 25% depth reduction with >90% original performance retained—outperforming state-of-the-art methods in accuracy, speed, and sustainability

Core Idea
- Identify redundant consecutive transformer blocks.
- Replace them with a single linear transformation estimated from a small calibration dataset.
- Fuse this linear transformation into the model's structure with no new parameters and no retraining.

Methodology

1. Layer Selection

We determine which transformer blocks to remove by analyzing the similarity of hidden states before and after each block. Blocks with the smallest cosine distance between activations are pruned. This helps identify the least important layers while maintaining stability:

$$i^* = \arg\min_i D\left(\mathbf{L}_i, \mathbf{L}_{i+n}\right),$$

where $D(\cdot)$ – various distance metrics, i – optimal cut index, $\mathbf{L}_i$ – output of the transformer block.

2. Linear Transformation Estimation

After removing blocks, a linear transformation matrix ($\mathbf{T}$) is computed to approximate the effect of the pruned blocks. This is done by comparing the input and output activations of the skipped layers on a small calibration dataset.

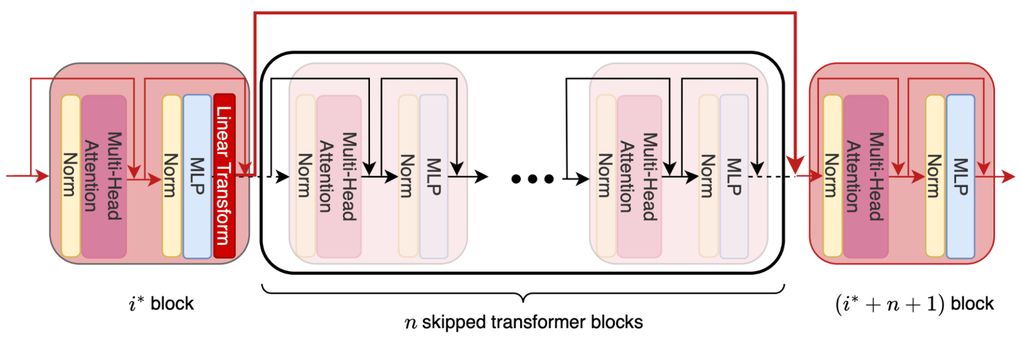


Fig. 1 ReplaceMe bypasses a sequence of transformer blocks using a learned linear transformation that maps outputs directly to later activations.

Formally, one needs to solve following optimization problem:

$$\mathbf{T}^* = \arg\min_{\mathbf{T}} h\left(\mathbf{M}_i \cdot \mathbf{T} + \mathbf{Y}_i; \mathbf{L}_{i+n}\right),$$

where $h(\cdot)$ – distance function, $\mathbf{M}_i$ and $\mathbf{Y}_i$ are outputs of MLP and attention sub-blocks in i^{th} transformer block.

Two approaches are used to find the best transformation:

- **Analytical (Least Squares)** – fast and deterministic ($h(\cdot) = \mathbf{L}_2$):

$$\mathbf{T}^* = \left(\mathbf{M}_i^T \cdot \mathbf{M}_i\right)^{-1} \cdot \mathbf{M}_i^T \cdot \left(\mathbf{L}_{i+n} - \mathbf{Y}_i\right),$$

- **Numerical (Cosine Optimization)** – slightly slower, but better approximation.

$$\mathbf{T}^* = \arg\min_{\mathbf{T}} \sum_{k=1}^{N} \left(1 - \frac{\left(\mathbf{M}_{i,k} \cdot \mathbf{T} + \mathbf{Y}_{i,k}\right)^{\mathsf{T}} \cdot \mathbf{L}_{i+n,k}}{\left\|\mathbf{M}_{i,k} \cdot \mathbf{T} + \mathbf{Y}_{i,k}\right\|_2 \left\|\mathbf{L}_{i+n,k}\right\|_2}\right),$$

where $\mathbf{M}_{i,k}$ – denotes k^{th} row of matrix M_i

The learned transformation is **fused** into the nearest MLP layer, so the **network architecture remains consistent**.

3. Extension: Multiple Linear Transforms and Robust Optimization

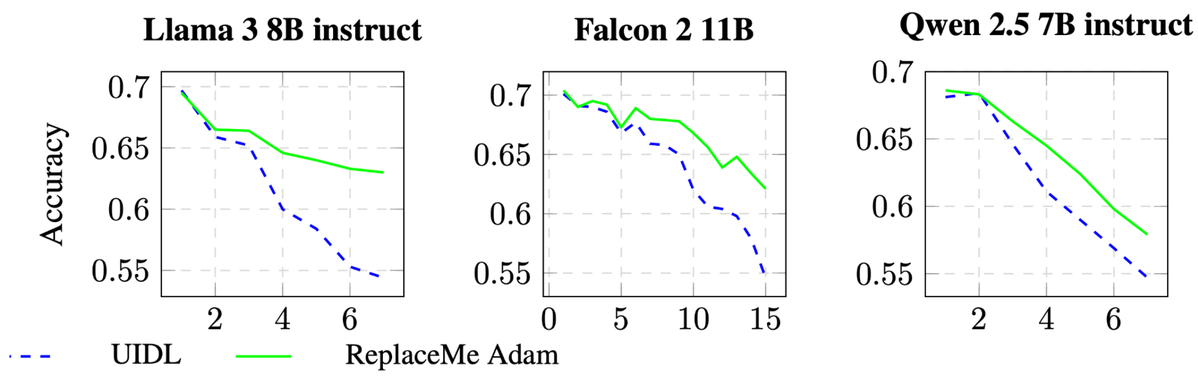
- To stabilize the transformation, L1 or L2 regularization is applied. A regularization coefficient (alpha) controls the trade-off between perplexity and accuracy.
- ReplaceMe can also use multiple non-overlapping linear transformations (Multi-LT). This allows more flexibility when pruning larger portions of the model.

Experiments

We evaluated ReplaceMe on several model families:
- LLaMA-2-7B, LLaMA-3-8B-Instruct
- Qwen2.5-7B, Falcon-11B
- CLIP-ViT for vision experiments

Calibration datasets: FineWeb, SlimOrca, and self-generated instruction data.
Benchmarks: CMNLI, MMLU, Winogrande, BoolQ, OpenBookQA, SciQ, etc..

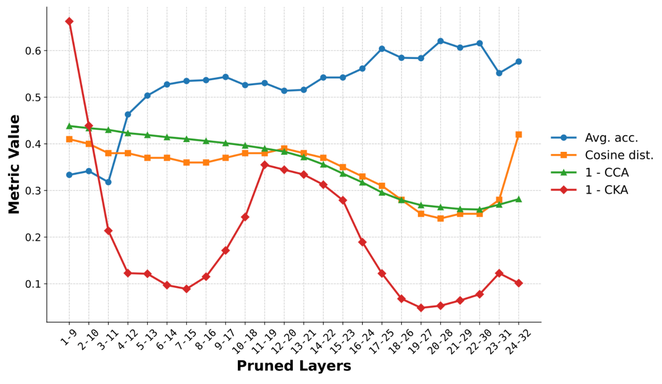
Method	Train-Free	C3	CMNLI	CHID (test)	WSC	Hella Swag	PIQA	Race-M	Race-H	MMLU	CMMLU	AVG	RP
Llama 2 7B (baseline)		43.8	33.0	41.6	37.5	71.3	78.1	33.1	35.5	46.8	31.8	45.3	100.0%
LLM-Streamline*	✗	**43.3**	33.0	24.1	36.5	**61.1**	**71.5**	34.8	37.0	45.5	29.4	41.6	92.0%
LLMPruner*	✗	29.7	33.4	28.4	40.4	54.6	72.0	22.9	22.0	25.3	25.0	35.4	78.2%
SliceGPT*	✗	31.5	31.6	18.5	43.3	47.5	68.3	27.0	29.4	28.8	24.8	35.1	77.5%
LaCo*	✗	39.7	**34.4**	**36.1**	40.4	55.7	69.8	23.6	22.6	26.5	25.2	37.4	82.7%
UIDL*	✗	40.2	**34.4**	21.5	40.4	59.7	69.0	35.2	34.7	44.6	28.9	40.9	90.3%
Ours (Cosine)	✓	42.5	33.0	25.2	38.5	59.4	71.1	35.4	**36.7**	**46.4**	**30.4**	**41.9**	**92.5%**
Ours (LS)	✓	39.4	33.0	18.9	38.5	58.5	70.5	**37.1**	36.5	45.2	29.2	40.7	89.9%



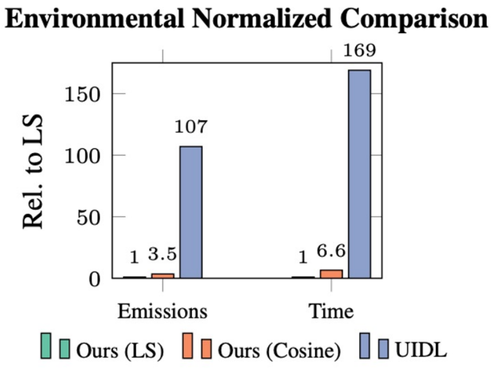
Method	Linear transform	Lambada-openai ppl ↓	Avg-acc↑	RP↑
Llama 3 8B Instruct [8]		3.11	0.7	100%
SVD-LLM [53]	None	29.90	0.59	85.3%
LLMPruner [29]	None	12.31	0.60	85.3%
UIDL [13]	Identity	2216.96	0.58	82.5%
ReplaceMe(ours)	Linear (LS)	20.23	0.63	89.9%
ReplaceMe(ours)	Linear (Cosine)	**15.88**	**0.63**	**90.9%**
ReplaceMe(ours)	Multi_LT_NC (Cosine)	**13.95**	**0.63**	90.0%

Model	Compres. ratio	MS-COCO Captions (retrieval)		Cifar10 (zero-shot)		VOC2007 Multilabel (zero-shot)	VTAB/EuroSAT	
		text recall@5	vision recall@5	acc1	acc5	mean_avg_p	acc1	acc5
CLIP-L/14 [37]	-	0.794	0.611	0.956	0.996	0.790	0.625	0.960
UIDL	13%	0.745	0.609	0.927	0.996	0.781	0.490	0.931
ReplaceMe (LS)	13%	**0.767**	**0.620**	**0.939**	0.996	**0.800**	**0.552**	**0.941**
UIDL	25%	0.515	0.418	0.693	0.971	0.597	0.381	0.814
ReplaceMe (LS)	25%	**0.556**	**0.471**	**0.780**	0.971	**0.688**	**0.395**	**0.823**

Layer Selection Strategy:



Environmental Impact:



Conclusion

Our proposed ReplaceMe:
- Allows training-free structured depth pruning
- Replaces redundant blocks with a simple Linear layer
- Maintains 90%+ accuracy at 25% compression
- Works across LLMs and Vision Transformers

Code is available here: github.com/mts-ai/ReplaceMe

References

Gromov et al. (2025). "The Unreasonable Ineffectiveness of the Deeper Layers" In: ICLR 2025
Razzhigaev et al. (2025). "Your Transformer is Secretly Linear" In: CoRR 2024