CORP: Closed-Form One-shot Representation-Preserving Structured Pruning

Boxiang Zhang
Purdue University
zhan4653@purdue.edu

Baijian Yang
Purdue University
byang@purdue.edu

Abstract

Transformers [Vaswani et al., 2017] achieve strong accuracy but incur high compute and memory cost. Structured pruning reduces inference cost, but most methods rely on retraining or multi-stage optimization, which limits post-training deployment. We propose **CORP**, a closed-form one-shot structured pruning method that removes MLP dimensions and attention substructures using only unlabeled calibration data without gradients or fine-tuning. CORP formulates structured pruning as a representation recovery problem. It models removed components as affine functions of retained components and derives closed-form ridge regression solutions that fold compensation into model weights. This minimizes a layer-local affine/logit reconstruction objective under the calibration distribution. Experiments on ImageNet [Russakovsky et al., 2015] with DeiT [Touvron et al., 2021] reveal strong redundancy in both MLP and attention representations. With CORP, models retain high accuracy under aggressive sparsity. On DeiT-Huge, CORP achieves 83.24% Top-1 accuracy after pruning 50% of both MLP and attention structures.

1 Introduction

Transformer-based models achieve strong performance across multimodal tasks but incur high computational and memory cost. These costs limit deployment on resource-constrained hardware. Structured pruning offers practical speedups by removing entire dimensions or substructures. However, most structured pruning methods rely on retraining or iterative optimization. In deployment settings, labels, gradients, and training access may not always be available. This motivates *post-training one-shot pruning*, where compression must complete in a single calibration pass without iterative optimization. Under this constraint, naive structured pruning fails, as removing channels or attention dimensions introduces systematic error that causes rapid accuracy collapse even at moderate sparsity.

We argue that this failure cannot be explained solely by ranking quality and is largely driven by uncorrected representation error. Structured pruning removes additive components of internal representations. Without explicit recovery, these errors accumulate across layers. These results suggest that compensation may be more important than increasingly complex importance metrics.

We propose **CORP**, a closed-form one-shot structured pruning framework for Transformers. CORP casts pruning as a representation recovery problem, removing MLP hidden dimensions and attention (Q/K) channels. It models removed components as affine functions of retained ones and derives closed-form ridge solutions from unlabeled calibration data. The compensation folds into weights producing standard linear layers with reduced dimensions without inference overhead.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

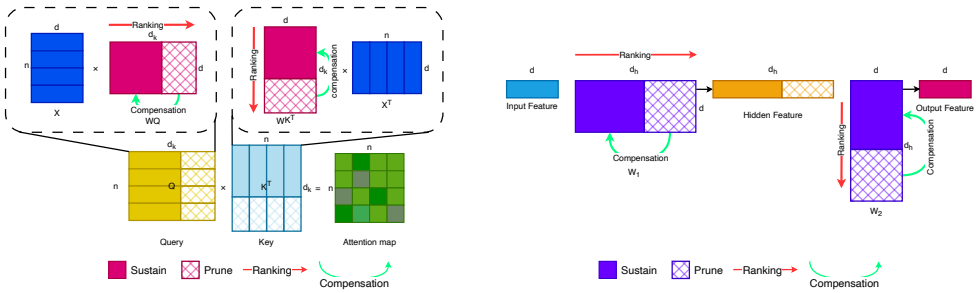


Figure 1: Illustration of CORP. Pruning removes structured components, while CORP reconstructs the missing contribution using closed-form compensation folded into the remaining weights. **(a)** Attention head dimension pruning removes channel dimensions in the query and key projections. **(b)** MLP structured pruning removes entire hidden dimensions between the two linear layers.

CORP retains strong accuracy at scale. For DeiT-Huge, it achieves 83.24% Top-1 at 50% sparsity with $>$40% FLOPs/parameter reduction and 1.6$\times$ speedup on ImageNet classification. On OPT-1.3B (30%), perplexity rises modestly (14.60$\rightarrow$18.53/19.36) with up to 17.03% FLOPs reduction on WikiText-2 [Merity et al., 2016]. On DINOv2, 50% backbone pruning preserves downstream performance, improving RMSE (0.330$\rightarrow$0.317). These results show strong one-shot structured pruning on vision models and provide preliminary evidence beyond image classification.

Our contributions are:

- We reframe one-shot structured pruning as a representation recovery problem, shifting focus from ranking to reconstruction, while retaining simple ranking criteria.
- We propose CORP, a closed-form compensation framework for one-shot structured pruning without labels, gradients, or fine-tuning.
- We demonstrate strong accuracy preservation and practical efficiency gains across scales.

2 Related Work

2.1 Post-Training Pruning

Traditional model compression — quantization-aware training [Wang et al., 2023, Dettmers et al., 2023], pruning with fine-tuning [Sanh et al., 2020, Zhang et al., 2024b], and knowledge distillation [Zhao et al., 2024, Gou et al., 2021, Jin et al., 2023] — requires labels, gradients, or repeated optimization. Post-training compression instead uses only a pretrained model and a small calibration set; methods differ in pruning scope, selection signal, and recovery strategy.

Unstructured Early post-training pruning focuses on unstructured pruning, which removes individual **weights**. Optimal Brain Damage (OBD) [LeCun et al., 1989] and Optimal Brain Surgeon (OBS) [Hassibi and Stork, 1992] estimate loss sensitivity via the Hessian and compensate the model during pruning. Post-training quantization also leverages second-order information. GPTQ [Frantar et al., 2023a] uses a Hessian approximation to minimize quantization error. Optimal Brain Compression (OBC) [Frantar et al., 2023b] applies layer-wise Gauss–Newton blocks to improve scalability and unify pruning and quantization under a second-order framework.

The computation of the Hessian and its inverse is the main bottleneck for scalability. WoodFisher [Singh and Alistarh, 2020] approximates the Hessian using empirical Fisher information with Woodbury updates. SparseGPT [Frantar and Alistarh, 2023] scales unstructured pruning to LLMs via layer-wise pruning with implicit curvature modeling. Despite improved accuracy recovery, second-order methods require estimating or inverting large curvature matrices. These costs grow rapidly with model width and depth. Under one-shot constraints, such computation and memory overhead limit scalability to large Transformers. Wanda [Sun et al., 2024] scales by eliminating the Hessian and ranking importance via the magnitude–activation products. While unstructured methods

Method	Pruning Scope	Selection Signal	Recovery	Finetune Free
SparseGPT	Weight/N:M weight	Hessian approx.	Second-order weight update	✓
OBC	Weight/quant. group	Gauss–Newton / OBS	Second-order weight update	✓
Wanda	Weight	Act. × Mag.	None	✓
OPTIN	MLP channel/attn head/token	Trajectory score	None	✓
Olica	MLP channel / attn head dim.	Act. × Mag. (FFN) / PCA (attn)	Low-rank ridge residual branch (FFN only)	✓
KCM	MLP channel	Output contribution	Scaling compensation	✓
VBP	MLP channel	Activation variance	Bias compensation	✗
DC-ViT	Attn module / MLP block	Recover-ability score	Iterative Feature-mimic fine-tuning	✗
GRAIL	MLP channel / attn head	Mag./Wanda/Gram selection	Post-hoc Gram-ridge reconstruction	✓
SNOWS	Mask-defined structure	Representation objective	Iterative Hessian-free recovery	✓
CORP	MLP channel / QK dim.	Act./weight/logit energy	Closed-form affine/logit compensation	✓

Table 1: Comparison of representative post-training pruning methods. We separate pruning scope, selection signal, recovery strategy, and post-pruning fine-tuning requirement. CORP combines MLP-channel and QK-dimension pruning with closed-form compensation.

can achieve high sparsity with strong accuracy retention, the resulting sparse patterns yield limited inference speedups on standard hardware.

Structured Pruning Structured pruning removes entire heads, channels, or tokens [Wang et al., 2024, Peng et al., 2024]. However, many structured pruning methods rely on iterative mask refinement, staged optimization, or retraining, which conflict with strict one-shot post-training constraints. Several works extend second-order objectives to structured pruning. Kwon et al. [2022] introduce Fisher-based mask variables for heads and channels. LPViT [Xu et al., 2024] applies block-structured pruning guided by a second-order criterion. While effective, these approaches still depend on curvature estimation and block-wise matrix inversion. To avoid explicit Hessian computation, OPTIN [Khaki and Plataniotis, 2024] use trajectory-based importance, and OFB [Ye et al., 2024] learns sparsity patterns via bi-mask optimization. LLM-oriented structured pruning differs from CORP in scope or recovery budget: FLAP [An et al., 2024] combines fluctuation-based global allocation with bias compensation, SliceGPT [Ashkboos et al., 2024] removes embedding dimensions via computational invariance, and LLM-Pruner [Ma et al., 2023] relies on gradient-based importance and LoRA recovery, while LPViT targets block-structured sparsity with a hardware-aware objective.

Recent work highlights the role of activation statistics. SmoothQuant [Xiao et al., 2024] rescales activations to suppress outliers, and AWQ [Lin et al., 2024] selects important channels by activation magnitude. Olica [He and Lin, 2025] prunes FFN neurons with a Wanda-style score and reduces each head's dimension by PCA of the $W_Q W_K^\top$ and $W_V W_O^\top$ products; its FFN error is recovered by a closed-form ridge fit kept as a low-rank branch at inference, with no attention compensation. TransKV [Xu et al., 2026] instead derives the truncation basis from data-driven keys and values. These results motivate activation-guided criteria under one-shot constraints. We call this strict single-pass, fine-tuning-free, post-training setting *one-shot pruning*.

DC-ViT [Zhang et al., 2024a] performs few-shot ViT compression by removing entire attention modules and pruning portions of the remaining MLP, with a small MLP retained to absorb the resulting representation gap. Recovery relies on progressive feature-mimicking finetuning (MSE against the original model) using a tiny labeled-free training set. VBP [Berisha et al., 2025] performs structured MLP pruning using activation statistics without gradients or Hessian, but relies on ten epochs of distillation fine-tuning to recover accuracy. KCM [Nova et al., 2023] ranks MLP channels by layer-wise output statistics and mutual representability without gradients or fine-tuning, then rescales the kept channels to reconstruct layer-wise outputs. SNOWS [Lucas and Mazumder, 2024] preserves deeper representations using Hessian-free second-order optimization. GRAIL [Tang et al., 2026] performs post-hoc linear reconstruction after structured compression. These methods also recover representations after pruning, but differ from CORP in pruning scope, recovery objective, optimization strategy, and finetuning dependency (Table 1).

2.2 Positioning

CORP differs from prior representation-recovery methods in three ways: it jointly prunes MLP channels and Q/K attention dimensions, derives a closed-form logit-space compensator for attention, and folds all compensation directly into reduced projections without iterative optimization or extra inference modules. SNOWS [Lucas and Mazumder, 2024] also performs one-shot post-training

pruning without retraining, but relies on Hessian-free second-order optimization to preserve deeper nonlinear representations. In contrast, CORP uses local closed-form ridge compensation and avoids gradients, Hessian-vector products, and iterative updates. Recent concurrent work, GRAIL [Tang et al., 2026], is closer in spirit. For MLP blocks, both GRAIL and CORP use linear reconstruction after structured pruning. However, GRAIL compensates only the second linear layer to recover module outputs, while CORP performs layer-wise representation recovery and additionally introduces Q/K logit-space compensation for attention-dimension pruning by approximating $Q_P K_P^\top$ with $Q_S M K_S^\top$ and folding the correction into the query and key projections. Olica also fits a closed-form ridge solution, but regresses the pruned FFN's output residual on the block input and keeps it as an extra low-rank branch; CORP predicts the pruned activations from the retained ones and folds the correction into the existing MLP and Q/K weights, adding no inference-time parameters. Thus, CORP unifies MLP-channel and Q/K-dimension pruning under a single closed-form compensation framework. Like L-OBS [Dong et al., 2017], AdaQuant [Hubara et al., 2021], OBC, GPTQ, SparseGPT, and KCM, CORP preserves each layer's input–output mapping from a small unlabeled calibration set; it differs in using closed-form compensation for MLP channels and Q/K dimensions.

3 Method

3.1 Empirical Motivation and Hypothesis

We empirically find that MLP blocks in DeiT-Base [Touvron et al., 2021] are highly over-parameterized. Detailed statistics are reported in Table 9 (Appendix A). However, naive structured pruning causes severe accuracy degradation. We hypothesize that structured pruning introduces a *systematic* error: the structured, mask-determined contribution that pruning deletes from a module output, namely $W_P x_P$ for an MLP block and $Q_P K_P^\top$ for an attention head (Sec. 3.2). This term is not the total prediction error but the exact residual created by the mask; objectives (2) and (5) minimize its expected squared norm, i.e., they recover each module's output rather than optimize the end-to-end task loss (see the decompositions in Eqs. (13) and (18)). Correcting this error preserves the original representation. Based on this hypothesis, we cast pruning as a representation approximation problem. We remove redundant channels and compensate the induced error with a lightweight affine transformation. This yields a closed-form correction that preserves intermediate representations. We treat structured pruning as representation recovery on top of importance-based selection.

3.2 Problem Setup and Notation

We consider one-shot structured pruning of a pretrained Transformer using a small calibration set $\mathcal{D} = \{x^{(i)}\}_{i=1}^N$. All expectations are estimated empirically over $\mathcal{D}$.

MLP consists of two linear layers. The first projects activations to a higher-dimensional space, and the second maps them back to the original dimension. Our goal is to reduce the hidden dimension. We model the second MLP layer as an affine map defined by $W \in \mathbb{R}^{d\times o}$ and $b \in \mathbb{R}^d$ where d is the feature dimension and o is the hidden dimension:

$$y = Wx + b, \quad W = [W_S \;\; W_P], \quad x = \begin{bmatrix} x_S \\ x_P \end{bmatrix}. \tag{1}$$

Structured pruning removes input channels by partitioning indices into kept set S and pruned set P. After pruning, only x_S is available. We seek compensated parameters $(\widehat{W}_S, \widehat{b})$ that minimize

$$\min_{\widehat{W}_S, \widehat{b}} \; \mathbb{E}_{x\sim\mathcal{D}} \big\| Wx + b - (\widehat{W}_S x_S + \widehat{b}) \big\|_2^2. \tag{2}$$

Attention For self-attention, let $X \in \mathbb{R}^{n\times d}$ denote the token matrix. Query and key projections are

$$Q = XW_Q + \mathbf{1}b_Q^\top, \quad K = XW_K + \mathbf{1}b_K^\top, \tag{3}$$

with $W_Q, W_K \in \mathbb{R}^{d\times d_h}$, where d_h is the head dimension. We prune by splitting head dimensions.

$$Q = [Q_S \; Q_P], K = [K_S \; K_P] \quad \Rightarrow \quad L = QK^\top = Q_S K_S^\top + Q_P K_P^\top. \tag{4}$$

We seek compensated parameters $(\widehat{Q}_S, \widehat{K}_S)$ that minimize

$$\min_{\widehat{Q}_S, \widehat{K}_s} \mathbb{E}_{x\sim\mathcal{D}} \left\| QK^\top - (\widehat{Q}_S \widehat{K}_S^\top) \right\|_F^2. \tag{5}$$

We restrict attention pruning to the **Query** and **Key** projections and do not prune the Value projection. In multi-head attention, pruning V is more coupled. Removing V dimensions changes the per-head features before concatenation. Recovering it requires joint modeling of the attention weights, per-head value features, and the post-concatenation output projection. That is beyond the scope of this study.

3.3 Ranking

We rank MLP channels using simple, data-driven metrics based on activation energy and weight magnitude. For each channel i, the activation energy is defined as $E_i = \mathbb{E}_{x\sim\mathcal{D}}\left[x_i^2\right]$, computed on the calibration set $\mathcal{D}$. We also consider the corresponding weight magnitude $|W{:}, i|_2$, capturing its impact on the layer output. Inspired by Wanda, we also evaluate a combined metric $E_i|W{:}, i|_2$. Channels are ranked by these scores and those with the lowest importance are pruned. We rank head dimensions by expected logit energy. For column j of Q and K, the energy is $s_j = \mathbb{E}\left[\|q_j\|_2^2 \, \|k_j\|_2^2\right]$. Dimensions with the smallest s_j are pruned.

3.4 Closed-form Compensation

MLP We predict pruned activations from retained ones and fold the correction into the remaining weights, compensating the output error introduced by pruning. The original output is

$$y = W_S x_S + W_P x_P + b. \tag{6}$$

We approximate pruned activations as an affine function of kept activations:

$$x_P \approx B x_S + c, \tag{7}$$

We estimate (B, c) via ridge regression:

$$\min_{B,c} \|X_P - (BX_S + c\mathbf{1}^\top)\|_F^2 + \lambda \|B\|_F^2, \tag{8}$$

The closed-form solution is:

$$B = \Sigma_{PS}(\Sigma_{SS} + \lambda I)^{-1}, \qquad c = \mu_P - B\mu_S. \tag{9}$$

$$\Sigma = \begin{bmatrix} \Sigma_{SS} & \Sigma_{SP} \\ \Sigma_{PS} & \Sigma_{PP} \end{bmatrix}, \quad \Sigma_{SS} = \mathbb{E}[(x_S - \mu_S)(x_S - \mu_S)^\top], \quad \Sigma_{PS} = \mathbb{E}[(x_P - \mu_P)(x_S - \mu_S)^\top]. \tag{10}$$

A full derivation of the affine predictor and its closed-form solution is provided in Appendix B.1.

We next bound the layer distortion; the full derivation appears in Appendix C.1. The distortion equals:

$$J_{\mathcal{D}}^\star \;=\; \mathrm{tr}\big(W_P \, \Sigma_{P|S} \, W_P^\top\big), \qquad \Sigma_{P|S} := \Sigma_{PP} - \Sigma_{PS} \, \Sigma_{SS}^\dagger \, \Sigma_{SP}, \tag{11}$$

Comparing to the uncompensated baseline $J_{\mathrm{uncomp}} = \mathrm{tr}(W_P \, \Sigma_{PP} \, W_P^\top) + \|W_P \mu_P\|_2^2$ yields:

$$J_{\mathrm{uncomp}} - J_{\mathcal{D}}^\star \;=\; \underbrace{\mathrm{tr}\big(W_P \, \Sigma_{PS} \, \Sigma_{SS}^\dagger \, \Sigma_{SP} \, W_P^\top\big)}_{\geq 0} \;+\; \underbrace{\|W_P \, \mu_P\|_2^2}_{\geq 0}, \tag{12}$$

so compensation strictly reduces the error whenever the kept activations carry any linear signal about the pruned ones in the directions of W_P, or the pruned channels have non-zero mean.

In practice CORP fits $(\widehat{B}_\lambda, \widehat{c}_\lambda)$ on a calibration distribution $\mathcal{D}$ from N samples and deploys on a distribution $\mathcal{D}_t$. The test-time error decomposes as

$$\mathcal{E}_t(\widehat{B}_\lambda, \widehat{c}_\lambda) \;\leq\; 2\, J_{\mathcal{D}_t}^\star \;+\; \mathcal{O}\big(|S|/N\big) \;+\; \mathcal{O}\big(\|\Sigma^{(c)} - \Sigma^{(t)}\|_{\mathrm{op}}\big), \tag{13}$$

The three terms isolate (i) the irreducible affine residual on $\mathcal{D}_t$, (ii) the calibration sample-complexity error, and (iii) the calibration–deployment shift. Equations (11)–(13) make the two roles of CORP explicit: ranking controls the magnitude of W_P, while compensation controls the residual covariance $\Sigma_{P|S}$. The method does not require globally low-rank activations; only the conditional residual $\Sigma_{P|S}$ in the directions of W_P must be small.

Attention To compensate for the missing logits, we approximate $Q_P K_P^\top \approx Q_S M K_S^\top$, where $M \in \mathbb{R}^{d'_h \times d'_h}$ models interaction between removed and kept logit subspaces for each head. For sample b in calibration set $\mathcal{D}$, we estimate M for each layer and attention head by solving

$$\min_M \sum_{b=1}^{N} \|Q_{P,b} K_{P,b}^\top - Q_{S,b} M K_{S,b}^\top\|_F^2 + \lambda \|M\|_F^2. \tag{14}$$

Vectorizing gives the closed-form ridge system

$$\left[\sum_{b=1}^{N} (K_{S,b}^\top K_{S,b}) \otimes (Q_{S,b}^\top Q_{S,b}) + \lambda I\right] \mathrm{vec}(M) = \sum_{b=1}^{N} \mathrm{vec}\left((Q_{S,b}^\top Q_{P,b})(K_{P,b}^\top K_{S,b})\right). \tag{15}$$

The derivation of the normal equations and logit-space compensation is given in Appendix B.2. Finally, we factor $I + M = U \Sigma V^\top$ and fold compensation into the projections:

$$\widehat{W}_{Q,S} = W_{Q,S} U \Sigma^{1/2}, \qquad \widehat{W}_{K,S} = W_{K,S} V \Sigma^{1/2}. \tag{16}$$

Let $\widehat{M}_\lambda$ be the ridge-compensated solution in (15), and substitute $G = \sum_b (K_{S,b}^\top K_{S,b}) \otimes (Q_{S,b}^\top Q_{S,b})$, $h = \sum_b \mathrm{vec}\big((Q_{S,b}^\top Q_{P,b})(K_{P,b}^\top K_{S,b})\big)$ into the objective gives the unregularized optimum distortion

$$J_{\mathcal{D}}^\star \;=\; \sum_{b=1}^{N} \|Q_{P,b} K_{P,b}^\top\|_F^2 \;-\; h^\top G^\dagger \, h, \tag{17}$$

where the first term is the uncompensated logit error obtained by $M = 0$ and the second is the strict compensation gain. The gain is non-negative, and equals zero only when the kept Q/K subspaces carry no bilinear signal about the pruned ones.

Under calibration–test shift, the test-time logit distortion decomposes as

$$\mathcal{E}_t(\widehat{M}_\lambda) \;\le\; 2\, J_{\mathcal{D}_t}^\star \;+\; \mathcal{O}\big(d_h'^2/N\big) \;+\; \mathcal{O}\big(\|G^{(c)}/N - G^{(t)}\|_{\mathrm{op}}\big), \tag{18}$$

The three terms have the same interpretation as in the MLP case, with the parameter dimension $d_h'^2$, the size of the per-head compensator M. A full derivation of the bound is provided in Appendix C.2. These bounds justify the local compensation objective for deployment data close to the calibration distribution, which Appendix F.1 verifies on held-out images and classes; they do not imply global optimality or robustness under arbitrary shift. This locality is a deliberate trade-off. Given the activations cached from one pass of the unpruned model, each objective is quadratic in its own compensator, so every MLP block and attention head admits a closed-form ridge solution and all solves run independently and in parallel, without labels, gradients, or sequential re-calibration (compensation is below 1% of total runtime, Table 6). Coupling modules through the intervening nonlinearities would make the objective non-quadratic and require iterative optimization, as in SNOWS [Lucas and Mazumder, 2024]. The price is that the residual e_ℓ left by each module is passed downstream unmodeled. The deviation of the final representation is bounded by a sum of the local residual norms weighted by the operator norms of the downstream Jacobians, as in layer-wise post-training compression [Dong et al., 2017, Hubara et al., 2021, Frantar et al., 2023b,a, Frantar and Alistarh, 2023, Nova et al., 2023]; minimizing each $\|e_\ell\|$ therefore controls, but does not eliminate, accumulated error, since the bound is an upper bound and deeper layers inherit the residuals of earlier ones.

Algorithm 1 summarizes the complete CORP pipeline. All statistics are collected in a single forward pass of the *unpruned* model: X_S, X_P in Eq. (8) and Q_S, Q_P, K_S, K_P in Eq. (14) are taken from the original network, with subscripts only indexing kept and pruned subsets. Each MLP layer and attention head is then solved independently from this one cache, which is what allows a single calibration pass and parallel solves; there is no joint or sequential optimization across modules.

4 Experiments

We evaluate pretrained DeiT [Touvron et al., 2021, 2022] and ViT [Dosovitskiy et al., 2021] on ImageNet [Russakovsky et al., 2015] validation set. For transferability, we prune pretrained DINOv2

Algorithm 1 CORP: One-shot Structured Pruning with Compensation

Input: Pretrained model f_θ, calibration set $\mathcal{D}$, MLP sparsity s_{mlp}, attention sparsity s_{attn}, ridge λ
Output: Pruned and compensated model $f_{\hat{\theta}}$
Run f_θ on $\mathcal{D}$ and cache MLP activations and attention Q, K
for each MLP block, each attention head **do**
  Rank hidden channels RANKMLP (Alg. 2)
  Rank Q/K dimensions RANKATTN (Alg. 4)
  affine compensation COMPENSATEMLP (Alg. 3)
  logit compensation COMPENSATEATTN (Alg. 5)
end for
return $f_{\hat{\theta}}$

	Base	MLP			Attn			Both		
Model	Top1 | FLOPs | P	Top1	FLOPs |↓	P	Top1	FLOPs |↓	P	Top1	FLOPs |↓	P
Tiny	72.02 | 1.4 | 5.7	55.43±0.05	1.1 | 24.3%	3.9	64.33±0.14	1.1 | 21.6%	5.3	41.69±0.24	0.8 | 45.9%	3.5
Small	79.72 | 5.0 | 22.1	69.35±0.12	3.6 | 28.1%	15.0	72.52±0.06	4.2 | 16.0%	20.3	58.47±0.26	2.8 | 44.1%	13.2
Base	81.74 | 18.3 | 86.6	72.20±0.16	12.7 | 30.5%	58.2	80.76±0.04	16.0 | 12.5%	79.5	70.17±0.14	10.4 | 43.0%	51.2
Large	84.58 | 63.5 | 304.4	82.06±0.06	43.7 | 31.2%	203.7	83.62±0.04	56.2 | 11.6%	279.2	80.34±0.06	36.3 | 42.8%	178.5
Huge	84.97 | 173.0 | 632.1	84.09±0.04	119.0 | 31.2%	422.3	84.22±0.03	153.0 | 11.7%	579.7	83.24±0.05	98.7 | 42.9%	369.9

Table 2: This table reports Top-1 accuracy (Top1), FLOPs in billions (G), and parameters in millions P under 50% structured sparsity. For pruned models, FLOPs are presented as 'FLOPs |↓", where ↓ indicates the percentage reduction relative to the corresponding base model. Top-1 of pruned models is reported as mean ± sample standard deviation over five independently sampled calibration sets; FLOPs and parameter counts are identical across these runs.

[Oquab et al., 2024] backbones and evaluate them on NYU Depth V2 [Nathan Silberman and Fergus, 2012] and ADE20k [Zhou et al., 2017]. For language modeling, we prune a pretrained OPT [Zhang et al., 2022] model using calibration samples from C4 [Dodge et al., 2021] and report perplexity on WikiText2 [Merity et al., 2016]. All experiments run on a single 24GB NVIDIA RTX 3090 GPU without kernel fusion or CUDA optimizations.

4.1 Results

Ranking & Calibration We compare activation-based, magnitude-based, and combined ranking strategies from Sec. 3.3, with and w/o compensation at 50% joint sparsity. The **combined strategy** achieves the best accuracy and is used by default. Ranking ablation results appear in Appendix E Figure 5. We also study the effect of calibration size at 50% joint pruning (Table 3). Sensitivity decreases with model scale: larger models remain stable even with small calibration sets, while smaller models benefit more from more calibration data. We use **4000** calibration samples by default. We use a single fixed ridge coefficient $\lambda = 10^{-3}$ for every MLP and attention solve, with no per-model, per-layer, or per-sparsity tuning; on DeiT-Large at 50–70% sparsity, accuracy varies by less than 0.06 points over $\lambda \in [10^{-5}, 10^{-3}]$ (Appendix F.2). Results are also stable across calibration sampling: over five independently drawn calibration sets, the sample standard deviation of Top-1 is at most 0.26 points for every pruned entry in Table 2, and at most 0.06 points for DeiT-L and DeiT-H.

DeiT Performance Table 2 reports Top-1 accuracy, FLOPs, and parameter counts for five DeiT model sizes under 50% sparsity. Attention pruning preserves accuracy better than MLP pruning at matched sparsity because attention contributes a smaller fraction of FLOPs (∼12% vs. ∼30%). Accuracy preservation also improves substantially with model scale: under 50% joint sparsity, DeiT-Huge drops only 1.7 points (84.97→83.24), while DeiT-Tiny collapses to 41.69, indicating that larger ViTs show stronger redundancy in our setting for pruning with closed-form compensation.

Accuracy vs Sparsity We further explore the effect of sparsity on accuracy for DeiT-Large and DeiT-Huge. Figure 2 shows Top-1 accuracy as a function of sparsity for DeiT-Large and DeiT-Huge. Without compensation, accuracy collapses at high sparsity, while compensation preserves stable performance across MLP, attention, and joint pruning. The benefit increases with sparsity; for

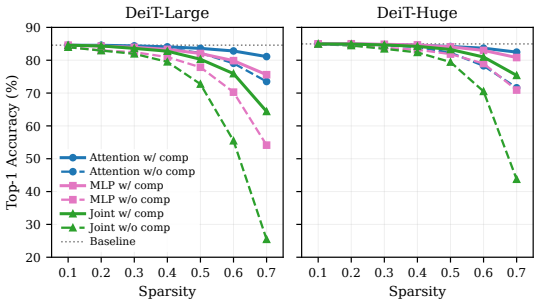


Figure 2: Top-1 ImageNet accuracy vs. sparsity.

Calib	Tiny	Small	Base	Large	Huge
100	38.75	57.24	61.06	78.86	82.46
500	39.96	57.47	63.13	79.37	82.89
1000	39.96	58.02	66.01	79.64	82.92
2000	40.96	58.19	68.36	80.26	83.08
4000	41.36	58.37	70.00	80.30	83.27

Table 3: Effect of calibration set size on Top-1 accuracy under 50% joint sparsity (both MLP and attention). DeiT-L and DeiT-H remain stable even with small calibration sets.

Method	Scope	Sparsity	Top-1	Δ
SNOWS	Attn	2:4	81.01	-3.16
GRAIL	Attn	50%	74.47	-11.37
CORP	Attn	50%	**81.98**	**-2.39**
SNOWS	MLP	2:4	69.71	-14.46
GRAIL	MLP	50%	76.13	-9.71
CORP	MLP	50%	**81.80**	**-2.58**

(a) CORP vs. SNOWS and GRAIL.

Method	FLOPs↓	Top-1	Δ
DC-ViT	8.3%	83.78	-0.63
CORP	11.6%	80.49	-0.60
DC-ViT	16.6%	83.15	-1.26
CORP	15.5%	79.80	-1.29
DC-ViT	24.8%	81.60	-2.81
CORP	23.5%	78.24	-2.86

(b) CORP vs. DC-ViT at matched FLOPs reduction.

Table 4: Comparison with representative structured pruning methods.

example, under 70% joint sparsity on DeiT-Huge, compensation improves accuracy from 43.8% to 75.4%. DeiT-Huge remains within 2.5% of the baseline up to 50% sparsity across all pruning targets. Appendix F.1 directly measures the recoverability ratios ρ^2 of Propositions C.1.2 and C.2.2 on DeiT-Large (50% joint): compensation recovers 47% (MLP) and 78% (attention) of the pruned energy on held-out images, and the oracle-to-transfer gap stays below 0.01 (pooled), including when calibration and evaluation classes are disjoint.

Benchmark We compare against **VBP** [Berisha et al., 2025], **DC-ViT** [Zhang et al., 2024a], **GRAIL** [Tang et al., 2026], and **SNOWS** [Lucas and Mazumder, 2024]. We exclude KCM [Nova et al., 2023] because its self-pretrained model is not publicly available. GRAIL results are reproduced using the official implementation under the same calibration setting as CORP; all other comparisons use reported settings from prior work and therefore differ in architecture, pruning scope, and dense-model baseline. We report relative accuracy to partially reduce this mismatch. We compare only against methods with matching models, pruning units (channels or head dimensions), and adaptation budget; block-sparse (LPViT) and LLM-specific methods (FLAP, SliceGPT, LLM-Pruner) are discussed in Sec. 2 because their sparsity is measured on a different axis.

Figure 3 compares CORP, GRAIL, and VBP under MLP-only pruning. CORP consistently outperforms VBP and remains competitive with or better than GRAIL, especially at higher sparsity. At 50% MLP sparsity on DeiT-Base, CORP achieves 72.0% Top-1 versus 68.6 for GRAIL and 66.4 for VBP. Figure 4 compares accuracy at matched FLOPs reduction. Since VBP and GRAIL prune only MLP channels, while CORP distributes pruning across both MLP and attention, CORP achieves higher accuracy at every FLOPs target across all DeiT scales.

Table 4a compares CORP with GRAIL and SNOWS on ViT-Large at $\sim$50% compression. Under matched dimension pruning, CORP outperforms GRAIL by 7.51 points on attention (81.98 vs. 74.47) and 5.67 points on MLP (81.80 vs. 76.13), highlighting the benefit of CORP's compensation. SNOWS uses 2:4 weight sparsity, which preserves feature dimensions and relies on sparse kernels, while CORP removes entire channels. Despite the more aggressive structured setting, CORP achieves higher accuracy on both attention (81.98 vs. 81.01) and MLP (81.80 vs. 69.71).

DC-ViT removes attention modules and prunes MLP using few-shot feature-mimicking finetuning. As shown in Table 4b at comparable FLOPs reduction levels on ViT-B, CORP achieves nearly identical accuracy drops without finetuning. CORP therefore matches a heavily finetuned method

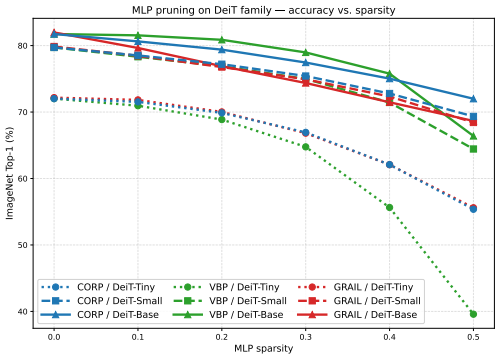


Figure 3: Comparison of CORP, VBP, and GRAIL on **MLP-only** pruning for DeiT family without fine-tuning at same MLP sparsity.

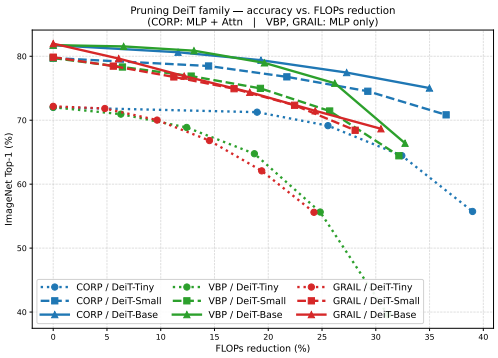


Figure 4: Pruning the DeiT family at matched FLOPs reduction. CORP prunes both MLP and attention; VBP and GRAIL prune only MLP.

Model	Sparsity	Top-1 (%)	Param (M)	FLOPs (G)	Lat (ms)	TP (fps)	Param↓ (%)	FLOPs↓ (%)	TP↑ (×)
DeiT-H	0.0	84.97	632.1	172.8	30.84	43	0.0	0.0	1.00
	0.1	84.96	579.7	153.7	27.70	48	8.3	11.1	1.12
	0.2	84.91	527.2	139.9	24.82	52	16.6	19.0	1.21
	0.3	84.66	474.8	126.2	22.73	57	24.9	27.0	1.32
	0.4	84.23	422.3	112.4	20.70	63	33.2	35.0	1.46
	0.5	83.27	369.9	98.7	19.16	71	41.5	42.9	1.64
	0.6	80.98	317.4	84.9	16.83	78	49.8	50.9	1.82
	0.7	75.38	265.0	71.2	14.71	92	58.1	58.8	2.13

Table 5: Accuracy and efficiency trade-offs of CORP on DeiT-Huge across sparsity levels. We report Top-1 accuracy, parameters (M), FLOPs (G), latency (ms), and throughput (fps). Latency and throughput are measured with batch size 1 and 16 respectively. Input resolution 224×224. Param↓ and FLOPs↓ denote reduction relative to the dense model, and TP↑ denotes throughput speedup.

within 0.05 ΔTop-1 using only a single calibration pass without gradients. Since the dense baselines differ (84.41 for DC-ViT ViT-B vs. 81.74 for CORP DeiT-B), ΔTop-1 is the more comparable metric.

Efficiency Table 5 reports accuracy, parameter count, FLOPs, p50 latency, and throughput for DeiT-Huge across sparsity levels from 0.0 to 0.7; results for all model sizes appear in Appendix G, Table 13. Structured sparsity enables real GPU speedups without custom kernels. At 50% sparsity, parameters and FLOPs drop by ∼40–45%, while throughput improves by ∼1.5× on DeiT-Base and ∼1.6× on DeiT-Huge. Throughput reaches 2–2.5× at 70–80% sparsity, though latency improves more gradually due to hardware and memory overheads. Table 6 shows that calibration dominates runtime, while ranking and compensation add negligible overhead.

Preliminary evidence beyond ImageNet. On **OPT-1.3B**, we followed Frantar and Alistarh [2023] and use 128 calibration sequences of length 2048 from the first shard of C4 [Dodge et al., 2021] and evaluate perplexity on WikiText-2 [Merity et al., 2016]. As shown in Table 7 at 30% sparsity, perplexity increases from 14.60 to 19.36 / 18.53 / 25.41 for MLP, attention, and joint pruning, respectively. The calibration–evaluation mismatch is only an initial test of robustness to distribution shift. CORP's cost is local to each layer and head, so its computation scales with depth and width; accuracy, however, is verified only up to OPT-1.3B, and we make no claim for larger LLMs.

On **DINOv2** [Oquab et al., 2024], we evaluate CORP on NYUv2 [Nathan Silberman and Fergus, 2012] depth estimation and ADE20K [Zhou et al., 2017] segmentation. We randomly sample 3,000 images from each dataset to form a 6,000-image unlabeled calibration set. We prune only the backbone to 50% sparsity and keep both task heads unchanged. As shown in Table 8, CORP preserves downstream performance. For DINOv2-Large, RMSE changes from 0.3492 to 0.3445, while δ_1 changes from 0.9466 to 0.9413. For DINOv2-Huge, RMSE improves from 0.3308 to 0.3167,

Model	P(M)	Cal	Rank	Comp	Tot
DeiT-T	5.7	81	0.02	0.40	81
DeiT-S	22.1	206	0.01	0.95	207
DeiT-B	86.6	567	0.02	3.22	570
DeiT-L	304.4	1831	0.05	13.70	1845
DeiT-H	632.1	4886	0.07	33.93	4920

Table 6: **P**: parameters (M); **Cal**: calibration; **Rank**: ranking; **Comp**: closed-form compensation; **Tot**: total runtime. Calibration dominates the total cost, while ranking and closed-form compensation introduce negligible overhead.

Target	PPL	FLOPs (T) / ↓	Params (B) / ↓
Baseline	14.60	6.19 / 0.0%	1.42 / 0.0%
MLP	19.36	5.20 / 16.0%	1.18 / 17.03%
Attn	18.53	5.50 / 11.2%	1.31 / 7.99%
Both	25.41	4.51 / 27.2%	1.06 / 25.01%

Table 7: Perplexity and efficiency of CORP on OPT-1.3B at 30% sparsity. **Target**: pruned component; **PPL**: perplexity; **FLOPs (T) / ↓**: FLOPs in trillions and reduction ratio; **Params (B) / ↓**: parameters in billions and reduction ratio.

Model	Params (M)	RMSE	δ_1	mIoU
Small	22.1 → 14.6	0.3992 → 0.4459	0.9192 → 0.8750	0.4198 → 0.2549
Base	86.6 → 56.9	0.3730 → 0.4201	0.9374 → 0.9118	0.4368 → 0.3527
Large	304.4 → 198.9	**0.3492** → **0.3445**	0.9466 → 0.9413	0.4313 → 0.4011
Huge	1.1B → 0.74B	**0.3308** → **0.3167**	0.9529 → 0.9514	0.4428 → 0.4367

Table 8: Parameter changes and downstream performance of structured-pruned DINOv2 on depth (RMSE, Δ_1) and segmentation (mIoU). Arrows denote baseline → pruned.

while δ_1 remains nearly unchanged. On segmentation, DINOv2-Huge shows only a small mIoU drop, from 0.4428 to 0.4367. CORP thus preserves DINOv2 backbone representations on both dense prediction tasks with a mixed calibration set, even though only the shared backbone is pruned and the task heads are unchanged.

5 Conclusion

We presented CORP, a one-shot structured pruning method that treats pruning as layer-wise representation recovery. CORP uses unlabeled calibration data to fit closed-form affine and logit-space compensation, then folds the correction into the retained weights. Experiments on DeiT show strong preservation on larger DeiT models and measurable throughput gains, while OPT and DINOv2 results provide preliminary evidence beyond ImageNet classification. CORP trades cross-module coupling for single-pass, closed-form, training-free compensation; future work should explore layer-wise sparsity allocation and stronger calibration–test distribution shift.

Limitations CORP compensates each MLP block and attention head independently from one set of calibration statistics of the unpruned model. This locality is a deliberate trade-off: it yields closed-form compensators from a single calibration pass, without labels, gradients, fine-tuning, or sequential re-calibration, and lets all solves run in parallel at negligible cost (Table 6). The price is that cross-module interactions are not modeled: residuals left by upstream modules are not re-fit downstream, so at high joint sparsity they can accumulate and reduce stability (Figure 2). The method uses a uniform sparsity ratio across layers and does not account for layer-wise redundancy differences. Because every module is solved independently, non-uniform per-layer ratios (e.g., guided by the effective ranks in Table 9) can be supplied without changing the formulation; rank-aware allocation is a natural extension. CORP targets strict post-training settings with no labels, gradients, or fine-tuning, and assumes calibration data aligned with deployment. Performance may degrade under strong calibration–test shift, very high sparsity, or architectures with weaker redundancy. Finally, CORP optimizes layer-wise representation recovery rather than end-to-end task loss and thus does not provide a global optimality guarantee. Compensation is computed in float32; we do not study its composition with quantization.

References

Yongqi An, Xu Zhao, Tao Yu, Ming Tang, and Jinqiao Wang. Fluctuation-based adaptive structured pruning for large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

Saleh Ashkboos, Maximilian L. Croci, Marcelo Gennari do Nascimento, Torsten Hoefler, and James Hensman. SliceGPT: Compress large language models by deleting rows and columns. In *International Conference on Learning Representations (ICLR)*, 2024.

Uranik Berisha, Jens Mehnert, and Alexandru Paul Condurache. Variance-based pruning for accelerating and compressing trained networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *arXiv preprint arXiv:2305.14314*, 2023.

Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner. Documenting large webtext corpora: A case study on the colossal clean crawled corpus. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1286–1305, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.98. URL https://aclanthology.org/2021.emnlp-main.98/.

Xin Dong, Shangyu Chen, and Sinno Jialin Pan. Learning to prune deep neural networks via layer-wise optimal brain surgeon. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021. URL https://arxiv.org/abs/2010.11929.

Elias Frantar and Dan Alistarh. SparseGPT: Massive language models can be accurately pruned in one-shot. *arXiv preprint arXiv:2301.00774*, 2023.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate post-training quantization for generative pre-trained transformers, 2023a. URL http://arxiv.org/abs/2210.17323.

Elias Frantar, Sidak Pal Singh, and Dan Alistarh. Optimal brain compression: A framework for accurate post-training quantization and pruning, 2023b. URL http://arxiv.org/abs/2208.11580.

Jianping Gou, Baosheng Yu, Stephen J. Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, March 2021. ISSN 1573-1405. doi: 10.1007/s11263-021-01453-z. URL http://dx.doi.org/10.1007/s11263-021-01453-z.

Babak Hassibi and David Stork. Second order derivatives for network pruning: Optimal brain surgeon. In *Advances in Neural Information Processing Systems*, volume 5. Morgan-Kaufmann, 1992. URL https://proceedings.neurips.cc/paper/1992/hash/303ed4c69846ab36c2904d3ba8573050-Abstract.html.

Jiujun He and Huazhen Lin. Olica: Efficient structured pruning of large language models without retraining. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, pages 22580–22594, 2025.

Roger A Horn and Charles R Johnson. *Matrix Analysis*. Cambridge University Press, Cambridge, 2013. ISBN 978-0-521-83940-2.

Daniel Hsu, Sham M. Kakade, and Tong Zhang. Random design analysis of ridge regression. *Found. Comput. Math.*, 14(3):569–600, June 2014. ISSN 1615-3375. doi: 10.1007/s10208-014-9192-1. URL https://doi.org/10.1007/s10208-014-9192-1.

Itay Hubara, Yury Nahshan, Yair Hanani, Ron Banner, and Daniel Soudry. Accurate post training quantization with small calibration sets. In *International Conference on Machine Learning (ICML)*, 2021.

Ying Jin, Jiaqi Wang, and Dahua Lin. Multi-level logit distillation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24276–24285. IEEE, 2023. ISBN 979-8-3503-0129-8. doi: 10.1109/CVPR52729.2023.02325. URL https://ieeexplore.ieee.org/document/10204863/.

Samir Khaki and Konstantinos N. Plataniotis. The need for speed: Pruning transformers with one recipe, 2024. URL http://arxiv.org/abs/2403.17921.

Woosuk Kwon, Sehoon Kim, Michael W. Mahoney, Joseph Hassoun, Kurt Keutzer, and Amir Gholami. A fast post-training pruning framework for transformers, 2022.

Yann LeCun, John Denker, and Sara Solla. Optimal brain damage. In *Advances in Neural Information Processing Systems*, volume 2. Morgan-Kaufmann, 1989. URL https://proceedings.neurips.cc/paper_files/paper/1989/hash/6c9882bbac1c7093bd25041881277658-Abstract.html.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. AWQ: Activation-aware weight quantization for LLM compression and acceleration, 2024. URL http://arxiv.org/abs/2306.00978.

Ryan Lucas and Rahul Mazumder. Preserving deep representations in one-shot pruning: A hessian-free second-order optimization framework, 2024. URL http://arxiv.org/abs/2411.18376. version: 1.

Xinyin Ma, Gongfan Fang, and Xinchao Wang. LLM-Pruner: On the structural pruning of large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models, 2016. URL https://arxiv.org/abs/1609.07843.

Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012.

Azade Nova, Hanjun Dai, and Dale Schuurmans. Gradient-free structured pruning with unlabeled data, 2023. URL http://arxiv.org/abs/2303.04185.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024. URL https://arxiv.org/abs/2304.07193.

Shuai Peng, Di Fu, Baole Wei, Yong Cao, Liangcai Gao, and Zhi Tang. Vote&mix: Plug-and-play token reduction for efficient vision transformer, 2024. URL http://arxiv.org/abs/2408.17062.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge, 2015. URL https://arxiv.org/abs/1409.0575.

Victor Sanh, Thomas Wolf, and Alexander M. Rush. Movement pruning: adaptive sparsity by fine-tuning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Sidak Pal Singh and Dan Alistarh. WoodFisher: Efficient second-order approximation for neural network compression, 2020. URL http://arxiv.org/abs/2004.14340.

G. W. Stewart. On the perturbation of pseudo-inverses, projections and linear least squares problems. *SIAM Rev.*, 19(4):634–662, October 1977. ISSN 0036-1445. doi: 10.1137/1019104. URL https://doi.org/10.1137/1019104.

Mingjie Sun, Zhuang Liu, Anna Bair, and J. Zico Kolter. A simple and effective pruning approach for large language models, 2024. URL https://arxiv.org/abs/2306.11695.

Wenwu Tang, Dong Wang, Lothar Thiele, and Olga Saukh. Grail: Post-hoc compensation by linear reconstruction for compressed networks, 2026. URL https://arxiv.org/abs/2602.23795.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers and distillation through attention, 2021. URL https://arxiv.org/abs/2012.12877.

Hugo Touvron, Matthieu Cord, and Hervé Jégou. Deit iii: Revenge of the vit, 2022. URL https://arxiv.org/abs/2204.07118.

Joel A. Tropp. An introduction to matrix concentration inequalities, 2015. URL https://arxiv.org/abs/1501.01571.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Hongjie Wang, Bhishma Dedhia, and Niraj K. Jha. Zero-TPrune: Zero-shot token pruning through leveraging of the attention graph in pre-trained transformers, 2024. URL http://arxiv.org/abs/2305.17328.

Hongyu Wang, Shuming Ma, Li Dong, Shaohan Huang, Huaijie Wang, Lingxiao Ma, Fan Yang, Ruiping Wang, Yi Wu, and Furu Wei. Bitnet: Scaling 1-bit transformers for large language models, 2023. URL https://arxiv.org/abs/2310.11453.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. SmoothQuant: Accurate and efficient post-training quantization for large language models, 2024. URL http://arxiv.org/abs/2211.10438.

Guangning Xu, Fanxu Meng, Ruijie Zhou, Michael K. Ng, Wenjie Pei, and Muhan Zhang. TransKV: A data-driven pruning method for large foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pages 2451–2461, 2026.

Kaixin Xu, Zhe Wang, Chunyun Chen, Xue Geng, Jie Lin, Mohamed M. Sabry Aly, Xulei Yang, Min Wu, Xiaoli Li, and Weisi Lin. LPViT: Low-power semi-structured pruning for vision transformers, 2024. URL http://arxiv.org/abs/2407.02068.

Hancheng Ye, Chong Yu, Peng Ye, Renqiu Xia, Yansong Tang, Jiwen Lu, Tao Chen, and Bo Zhang. Once for both: Single stage of importance and sparsity search for vision transformer compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5578–5588, June 2024.

Hanxiao Zhang, Yifan Zhou, and Guo-Hua Wang. Dense vision transformer compression with few samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15825–15834, 2024a.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. Opt: Open pre-trained transformer language models, 2022. URL https://arxiv.org/abs/2205.01068.

Yuxin Zhang, Lirui Zhao, Mingbao Lin, Yunyun Sun, Yiwu Yao, Xingjia Han, Jared Tanner, Shiwei Liu, and Rongrong Ji. Dynamic sparse no training: Training-free fine-tuning for sparse llms, 2024b. URL https://arxiv.org/abs/2310.08915.

Tianyang Zhao, Kunwar Yashraj Singh, Srikar Appalaraju, Peng Tang, Vijay Mahadevan, R. Manmatha, and Ying Nian Wu. No head left behind - multi-head alignment distillation for transformers. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press, 2024. ISBN 978-1-57735-887-9. doi: 10.1609/aaai.v38i7.28583. URL https://doi.org/10.1609/aaai.v38i7.28583.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

A Additional Analysis of MLP Redundancy in DeiT-Base

Table 9 summarizes an empirical analysis of MLP activation redundancy in **DeiT-Base**, whose MLP hidden dimension is 3072. The calibration set consists of 4000 images randomly selected from the ImageNet training set [Russakovsky et al., 2015]. Statistics are computed from calibration data collected at the output of each MLP activation layer.

First, MLP activations in DeiT-Base exhibit consistently low effective rank. Despite a hidden dimension of 3072, the effective rank rarely exceeds 350 across all transformer blocks. The effective rank ratio remains below 0.12 for every layer and drops below 0.06 in several cases. This indicates that MLP activations lie in a low-dimensional subspace.

Second, activation energy is highly concentrated. The number of channels required to explain 95% of the activation variance (k_{95}) is substantially smaller than the full feature dimension. In early layers, fewer than 20% of channels account for most of the activation energy. Even in later layers, strong energy concentration persists despite increased sparsity.

Third, activation sparsity increases with network depth. Later MLP blocks contain a large fraction of rarely active or mostly zero channels. In some layers, activation sparsity exceeds 0.9. These channels contribute minimally to forward computation and provide limited expressive power.

Fourth, redundancy is not uniform across layers. Early and middle layers exhibit low effective rank with moderate sparsity, while later layers combine low rank with extreme sparsity. This suggests that multiple redundancy mechanisms coexist across depth in DeiT-Base.

Taken together, these observations show that DeiT-Base MLP blocks contain substantial structured redundancy. However, directly removing channels introduces systematic bias rather than random noise. This explains why naive structured pruning leads to severe accuracy degradation. Effective pruning therefore requires explicit compensation to preserve the original function.

Layer	Dim	Eff. Rank	Rank Ratio	k_{95}	k_{95} Ratio	Act. Sparsity
blocks.0.mlp.act	3072	164.2	0.053	542	0.176	0.23
blocks.1.mlp.act	3072	136.4	0.044	336	0.109	0.30
blocks.2.mlp.act	3072	294.2	0.096	1118	0.364	0.16
blocks.3.mlp.act	3072	341.0	0.111	1577	0.513	0.08
blocks.4.mlp.act	3072	353.8	0.115	1804	0.587	0.06
blocks.5.mlp.act	3072	239.6	0.078	1783	0.580	0.07
blocks.6.mlp.act	3072	279.1	0.091	1878	0.611	0.08
blocks.7.mlp.act	3072	315.3	0.103	2037	0.663	0.14
blocks.8.mlp.act	3072	310.5	0.101	2009	0.654	0.59
blocks.9.mlp.act	3072	224.5	0.073	1617	0.526	0.86
blocks.10.mlp.act	3072	185.9	0.061	1203	0.392	0.92
blocks.11.mlp.act	3072	249.6	0.081	1646	0.536	0.19

Table 9: Empirical analysis of MLP activation redundancy across transformer blocks in DeiT-Base. Effective rank and k_{95} quantify energy concentration in the activation covariance. Rank ratios remain low across all layers, while activation sparsity increases in deeper blocks.

B Closed-Form Derivations

B.1 MLP Affine Compensation Derivation

Let $x \in \mathbb{R}^d$ be the layer input and split indices into kept set S and pruned set P: $x = [x_S^\top \; x_P^\top]^\top$. We fit an affine predictor

$$x_P \approx Bx_S + c, \tag{19}$$

where $B \in \mathbb{R}^{|P|\times|S|}$ and $c \in \mathbb{R}^{|P|}$. Let $X_S \in \mathbb{R}^{|S|\times N}$ and $X_P \in \mathbb{R}^{|P|\times N}$ stack N calibration samples. The compensated output becomes

$$\hat{y} = (W_S + W_P B)x_S + (b + W_P c) = \widehat{W}_S x_S + \widehat{b} \tag{20}$$

The ridge objective is

$$\min_{B,c} \|X_P - (BX_S + c\mathbf{1}^\top)\|_F^2 + \lambda\|B\|_F^2, \tag{21}$$

with $\mathbf{1} \in \mathbb{R}^N$.

Centering. Let $\mu_S = \frac{1}{N}X_S\mathbf{1}$ and $\mu_P = \frac{1}{N}X_P\mathbf{1}$. Define centered matrices $\bar{X}_S = X_S - \mu_S\mathbf{1}^\top$ and $\bar{X}_P = X_P - \mu_P\mathbf{1}^\top$. Optimizing over c yields

$$c = \mu_P - B\mu_S. \tag{22}$$

Substituting Eq. (22) into Eq. (21) reduces the problem to

$$\min_{B} \|\bar{X}_P - B\bar{X}_S\|_F^2 + \lambda\|B\|_F^2. \tag{23}$$

Closed form. Setting the gradient to zero gives

$$B(\bar{X}_S\bar{X}_S^\top + \lambda I) = \bar{X}_P\bar{X}_S^\top, \tag{24}$$

hence

$$B = \bar{X}_P\bar{X}_S^\top(\bar{X}_S\bar{X}_S^\top + \lambda I)^{-1}. \tag{25}$$

Using second-moment notation,

$$\Sigma = \begin{bmatrix}\Sigma_{SS} & \Sigma_{SP} \\ \Sigma_{PS} & \Sigma_{PP}\end{bmatrix}, \quad \Sigma_{SS} = E[(x_S - \mu_S)(x_S - \mu_S)^\top], \quad \Sigma_{PS} = E[(x_P - \mu_P)(x_S - \mu_S)^\top], \tag{26}$$

the same solution can be written as

$$B = \Sigma_{PS}(\Sigma_{SS} + \lambda I)^{-1}, \qquad c = \mu_P - B\mu_S. \tag{27}$$

B.2 Attention Logit Compensation Derivation

For clarity, we first derive the single-sample objective. In practice, CORP fits one compensation matrix per layer and head using all calibration samples. The calibration-set objective sums the single-sample losses, which yields a summed linear system over samples. We derive the Sylvester equation used in Section 3.4. Let $Q = [Q_S\ Q_P]$ and $K = [K_S\ K_P]$ denote the query and key matrices, where $Q_S, K_S \in \mathbb{R}^{n\times d_h'}$. The original logits decompose as:

$$L = QK^\top = Q_SK_S^\top + Q_PK_P^\top. \tag{28}$$

The missing logits after pruning are

$$T = Q_PK_P^\top. \tag{29}$$

We approximate T using the kept subspace:

$$T \approx Q_SMK_S^\top, \tag{30}$$

and logits using only kept dimensions:

$$\hat{L} = Q_S(I + M)K_S^\top, \tag{31}$$

where $M \in \mathbb{R}^{d_h'\times d_h'}$. We estimate M by ridge regression:

$$\min_{M} \|T - Q_SMK_S^\top\|_F^2 + \lambda\|M\|_F^2. \tag{32}$$

Normal equations. Let $A = Q_S$ and $B = K_S$. Then Eq. (32) becomes

$$\min_{M} \|T - AMB^\top\|_F^2 + \lambda\|M\|_F^2. \tag{33}$$

Define

$$\mathcal{L}(M) = \|T - AMB^\top\|_F^2 + \lambda\|M\|_F^2. \tag{34}$$

Using $\|X\|_F^2 = \mathrm{Tr}(X^\top X)$, we have

$$\mathcal{L}(M) = \mathrm{Tr}\left[(T - AMB^\top)^\top(T - AMB^\top)\right] + \lambda\mathrm{Tr}(M^\top M). \tag{35}$$

Expanding the trace gives

$$\mathcal{L}(M) = \mathrm{Tr}(T^\top T) - 2\mathrm{Tr}(T^\top AMB^\top) + \mathrm{Tr}(BM^\top A^\top AMB^\top) + \lambda \mathrm{Tr}(M^\top M). \tag{36}$$

taking the gradient

$$\nabla_M \mathcal{L}(M) = -2A^\top TB + 2A^\top AMB^\top B + 2\lambda M. \tag{37}$$

Setting $\nabla_M \mathcal{L}(M) = 0$ gives

$$(A^\top A)M(B^\top B) + \lambda M = A^\top TB. \tag{38}$$

Thus, we avoid constructing the full $n \times n$ matrix T. All terms in Eq. (38) have size $d'_h \times d'_h$.

Substituting $A = Q_S$, $B = K_S$ and $T = Q_P K_P^\top$ into Eq. (38) gives

$$(Q_S^\top Q_S)M(K_S^\top K_S) + \lambda M = (Q_S^\top Q_P)(K_P^\top K_S). \tag{39}$$

Thus, we avoid constructing the full $n \times n$ matrix T. All terms in Eq. (38) have size $d'_h \times d'_h$. This equation determines the compensation matrix M.

Using the identity

$$\mathrm{vec}\left((A^\top A)M(B^\top B)\right) = \left[(B^\top B) \otimes (A^\top A)\right] \mathrm{vec}(M). \tag{40}$$

we obtain

$$\left[(K_S^\top K_S) \otimes (Q_S^\top Q_S) + \lambda I\right] \mathrm{vec}(M) = \mathrm{vec}\left((Q_S^\top Q_P)(K_P^\top K_S)\right). \tag{41}$$

Therefore,

$$\mathrm{vec}(M) = \left[(K_S^\top K_S) \otimes (Q_S^\top Q_S) + \lambda I\right]^{-1} \mathrm{vec}\left((Q_S^\top Q_P)(K_P^\top K_S)\right). \tag{42}$$

Finally,

$$M = \mathrm{mat}\left(\left[(K_S^\top K_S) \otimes (Q_S^\top Q_S) + \lambda I\right]^{-1} \mathrm{vec}\left((Q_S^\top Q_P)(K_P^\top K_S)\right)\right), \tag{43}$$

where $\mathrm{mat}(\cdot)$ reshapes the vector into a $d'_h \times d'_h$ matrix.

For a calibration set, the exact normal equation becomes

$$\sum_{b=1}^{N} (Q_{S,b}^\top Q_{S,b})M(K_{S,b}^\top K_{S,b}) + \lambda M = \sum_{b=1}^{N} (Q_{S,b}^\top Q_{P,b})(K_{P,b}^\top K_{S,b}). \tag{44}$$

Equivalently,

$$\left[\sum_{b=1}^{N} (K_{S,b}^\top K_{S,b}) \otimes (Q_{S,b}^\top Q_{S,b}) + \lambda I\right] \mathrm{vec}(M) = \sum_{b=1}^{N} \mathrm{vec}\left((Q_{S,b}^\top Q_{P,b})(K_{P,b}^\top K_{S,b})\right). \tag{45}$$

In practice, CORP solves the calibration-summed system in Eq. (44)(45). The single-sample Sylvester form (Eq. (39)) is used only for derivation.

Folding compensation into projections. We seek $A, B \in \mathbb{R}^{d'_h \times d'_h}$ such that $AB^\top \approx I + M$ and update $\hat{W}_{Q,S} = W_{Q,S}A$ and $\hat{W}_{K,S} = W_{K,S}B$. We use the SVD $I + M = U\Sigma V^\top$ and set

$$A = U\Sigma^{1/2}, \quad B = V\Sigma^{1/2}, \tag{46}$$

which yields $AB^\top = U\Sigma V^\top = I + M$. This preserves the reduced head dimension while compensating missing logits in expectation.

C Layer Distortion Bounds and the Effect of Calibration Shift

C.1 MLP

We analyze the layer distortion induced by structured channel pruning under the ridge-regularized affine compensator used by CORP, and quantify how calibration–test distribution shift enters the bound. for a linear layer:

$$y = Wx + b, \qquad W = [\,W_S \;\; W_P\,], \qquad x = \begin{bmatrix} x_S \\ x_P \end{bmatrix}, \tag{47}$$

indices are partitioned into a kept set S and a pruned set P. After pruning, only x_S is available. Throughout this appendix, expectations are taken under a calibration distribution $\mathcal{D}$ unless stated otherwise. We write

$$\mu_S = \mathbb{E}_{\mathcal{D}}[x_S], \quad \mu_P = \mathbb{E}_{\mathcal{D}}[x_P], \quad \bar{x}_S = x_S - \mu_S, \quad \bar{x}_P = x_P - \mu_P, \tag{48}$$

and

$$\Sigma_{SS} = \mathbb{E}_{\mathcal{D}}[\bar{x}_S \bar{x}_S^\top], \quad \Sigma_{PP} = \mathbb{E}_{\mathcal{D}}[\bar{x}_P \bar{x}_P^\top], \quad \Sigma_{PS} = \mathbb{E}_{\mathcal{D}}[\bar{x}_P \bar{x}_S^\top], \quad \Sigma_{SP} = \Sigma_{PS}^\top. \tag{49}$$

The compensated parameters from Eqs. (8)–(10) of the main text, with ridge regularization $\lambda \geq 0$, are

$$\widehat{B}_\lambda = \Sigma_{PS}\,(\Sigma_{SS} + \lambda I)^\dagger, \qquad \widehat{c}_\lambda = \mu_P - \widehat{B}_\lambda \mu_S, \tag{50}$$

$$\widehat{W}_S = W_S + W_P \widehat{B}_\lambda, \qquad \widehat{b} = b + W_P \widehat{c}_\lambda, \tag{51}$$

where $(\,\cdot\,)^\dagger$ denotes the Moore–Penrose pseudoinverse. The layer distortion under the calibration distribution is

$$J_{\mathcal{D}}(\widehat{B}_\lambda, \widehat{c}_\lambda) \;=\; \mathbb{E}_{\mathcal{D}}\Big[\, \big\| W x + b - (\widehat{W}_S x_S + \widehat{b}) \big\|_2^2 \,\Big]. \tag{52}$$

Proposition C.1.1 (exact MLP layer error) *For any $\lambda \geq 0$, the ridge-compensated layer distortion in* (52) *equals*

$$J_{\mathcal{D}}(\widehat{B}_\lambda, \widehat{c}_\lambda) \;=\; \mathrm{tr}\Big(W_P\, \Sigma_\lambda\, W_P^\top\Big), \tag{53}$$

where

$$\Sigma_\lambda \;:=\; \Sigma_{PP} \;-\; \Sigma_{PS}\,(\Sigma_{SS} + \lambda I)^\dagger\, \Sigma_{SP} \;-\; \Sigma_{PS}\, \Sigma_{SP}\,(\Sigma_{SS} + \lambda I)^\dagger \;+\; \widehat{B}_\lambda\, \Sigma_{SS}\, \widehat{B}_\lambda^\top. \tag{54}$$

In the population limit $\lambda \to 0^+$ (or whenever Σ_{SS} is invertible), this collapses to

$$J_{\mathcal{D}}^\star \;=\; \mathrm{tr}\Big(W_P\, \Sigma_{P|S}\, W_P^\top\Big), \qquad \Sigma_{P|S} \;:=\; \Sigma_{PP} - \Sigma_{PS}\, \Sigma_{SS}^\dagger\, \Sigma_{SP}, \tag{55}$$

the residual covariance of x_P after the best linear prediction from x_S.

Proof. We proceed in four steps.

Step 1 (residual form). Substituting (51) into the residual gives, for any x,

$$\begin{aligned} W x + b - (\widehat{W}_S x_S + \widehat{b}) &= W_S x_S + W_P x_P + b - \big[(W_S + W_P \widehat{B}_\lambda) x_S + b + W_P \widehat{c}_\lambda\big] \\ &= W_P\,\big(x_P - \widehat{B}_\lambda x_S - \widehat{c}_\lambda\big). \end{aligned} \tag{56}$$

That is, the layer residual factors through W_P and the affine prediction residual of x_P from x_S.

Step 2 (centering the prediction residual). Define

$$r \;:=\; x_P - \widehat{B}_\lambda x_S - \widehat{c}_\lambda \;=\; \bar{x}_P - \widehat{B}_\lambda \bar{x}_S + \big(\mu_P - \widehat{B}_\lambda \mu_S - \widehat{c}_\lambda\big). \tag{57}$$

By the definition of $\widehat{c}_\lambda$ in (50), the parenthesized term vanishes, so

$$r \;=\; \bar{x}_P - \widehat{B}_\lambda \bar{x}_S, \qquad \mathbb{E}_{\mathcal{D}}[r] = 0. \tag{58}$$

Step 3 (residual covariance). Combining (56) and (58),

$$J_{\mathcal{D}}(\widehat{B}_\lambda, \widehat{c}_\lambda) = \mathbb{E}_{\mathcal{D}}\big[\, \|W_P r\|_2^2 \,\big] = \mathbb{E}_{\mathcal{D}}\big[\, \mathrm{tr}(W_P\, r r^\top\, W_P^\top) \,\big] = \mathrm{tr}\big(W_P\, \mathbb{E}_{\mathcal{D}}[r r^\top]\, W_P^\top\big). \tag{59}$$

The covariance of r expands as

$$\begin{aligned} \mathbb{E}_{\mathcal{D}}[r r^\top] &= \mathbb{E}_{\mathcal{D}}\Big[(\bar{x}_P - \widehat{B}_\lambda \bar{x}_S)(\bar{x}_P - \widehat{B}_\lambda \bar{x}_S)^\top\Big] \\ &= \Sigma_{PP} - \Sigma_{PS}\, \widehat{B}_\lambda^\top - \widehat{B}_\lambda\, \Sigma_{SP} + \widehat{B}_\lambda\, \Sigma_{SS}\, \widehat{B}_\lambda^\top. \end{aligned} \tag{60}$$

Substituting $\widehat{B}_\lambda = \Sigma_{PS}(\Sigma_{SS} + \lambda I)^\dagger$ gives Σ_λ as stated in (53).

Step 4 (population limit). When Σ_{SS} is invertible (or as $\lambda \to 0^+$ on the column space of Σ_{SS}), the optimality condition $\widehat{B}_0\, \Sigma_{SS} = \Sigma_{PS}$ implies

$$\widehat{B}_0\, \Sigma_{SS}\, \widehat{B}_0^\top = \Sigma_{PS}\, \Sigma_{SS}^\dagger\, \Sigma_{SP} = \widehat{B}_0\, \Sigma_{SP} = \Sigma_{PS}\, \widehat{B}_0^\top, \tag{61}$$

so (60) simplifies to

$$\mathbb{E}_{\mathcal{D}}[r r^\top] = \Sigma_{PP} - \Sigma_{PS}\, \Sigma_{SS}^\dagger\, \Sigma_{SP} = \Sigma_{P|S}, \tag{62}$$

which is the Schur complement of Σ on the kept block, yielding (55). □

Interpretation. The minimum layer distortion equals the trace of the Schur complement $\Sigma_{P|S}$ weighted by the removed columns W_P. Pruning is loss-free in the affine family iff $W_P\,\Sigma_{P|S}^{1/2} = 0$, i.e., the removed columns lie in the kernel of the residual covariance of x_P given x_S. Statistically, $\Sigma_{P|S}$ is the conditional covariance of x_P given a linear function of x_S; the result does not require x itself to be low-rank, only that the pruned activations be linearly predictable from the kept ones in the directions selected by W_P.

Proposition C.1.2 (MLP compensation gain) *Define the uncompensated layer distortion*

$$J_{\text{uncomp}} \;:=\; \mathbb{E}_{\mathcal{D}}\Big[\,\big\|Wx + b - (W_S x_S + b)\big\|_2^2\,\Big] \;=\; \operatorname{tr}\big(W_P\,\Sigma_{PP}\,W_P^\top\big) + \big\|W_P\,\mu_P\big\|_2^2. \tag{63}$$

Then in the population limit ($\lambda \to 0^+$), the compensation gain is

$$J_{\text{uncomp}} - J_{\mathcal{D}}^\star \;=\; \underbrace{\operatorname{tr}\big(W_P\,\Sigma_{PS}\,\Sigma_{SS}^\dagger\,\Sigma_{SP}\,W_P^\top\big)}_{\textit{variance-explained term} \,\geq\, 0} \;+\; \underbrace{\big\|W_P\,\mu_P\big\|_2^2}_{\textit{bias-correction term} \,\geq\, 0}, \tag{64}$$

with equality (zero gain) iff $W_P\,\mu_P = 0$ and $W_P\,\Sigma_{PS} = 0$. Equivalently, defining

$$\rho_{W_P}^2 \;:=\; \frac{\operatorname{tr}\big(W_P\,\Sigma_{PS}\,\Sigma_{SS}^\dagger\,\Sigma_{SP}\,W_P^\top\big)}{\operatorname{tr}\big(W_P\,\Sigma_{PP}\,W_P^\top\big)} \;\in\; [0,1], \tag{65}$$

the centered ratio satisfies $J_{\mathcal{D}}^\star \big/ \big(J_{\text{uncomp}} - \|W_P\mu_P\|_2^2\big) = 1 - \rho_{W_P}^2$.

Proof. We proceed in three steps.

Step 1 (uncompensated error). The uncompensated residual is $Wx + b - (W_S x_S + b) = W_P x_P$. Hence

$$J_{\text{uncomp}} = \mathbb{E}_{\mathcal{D}}\big[\,\|W_P x_P\|_2^2\,\big] = \operatorname{tr}\Big(W_P\,\mathbb{E}_{\mathcal{D}}[x_P x_P^\top]\,W_P^\top\Big). \tag{66}$$

Using $\mathbb{E}[x_P x_P^\top] = \Sigma_{PP} + \mu_P \mu_P^\top$,

$$\begin{aligned} J_{\text{uncomp}} &= \operatorname{tr}\big(W_P \Sigma_{PP} W_P^\top\big) + \operatorname{tr}\big(W_P \mu_P \mu_P^\top W_P^\top\big) \\ &= \operatorname{tr}\big(W_P \Sigma_{PP} W_P^\top\big) + \|W_P \mu_P\|_2^2. \end{aligned} \tag{67}$$

Step 2 (subtract the compensated optimum). By Proposition C.1.1 in the population limit, $J_{\mathcal{D}}^\star = \operatorname{tr}\big(W_P\,\Sigma_{PP}\,W_P^\top\big) - \operatorname{tr}\big(W_P\,\Sigma_{PS}\,\Sigma_{SS}^\dagger\,\Sigma_{SP}\,W_P^\top\big)$. Subtracting from (67),

$$J_{\text{uncomp}} - J_{\mathcal{D}}^\star = \operatorname{tr}\big(W_P\,\Sigma_{PS}\,\Sigma_{SS}^\dagger\,\Sigma_{SP}\,W_P^\top\big) + \|W_P \mu_P\|_2^2. \tag{68}$$

Step 3 (non-negativity). The matrix $\Sigma_{PS}\,\Sigma_{SS}^\dagger\,\Sigma_{SP}$ is positive semidefinite: writing $M := \Sigma_{SS}^{\dagger/2}\,\Sigma_{SP}$ (well-defined on $\operatorname{range}(\Sigma_{SS})$),

$$\Sigma_{PS}\,\Sigma_{SS}^\dagger\,\Sigma_{SP} = M^\top M \;\succeq\; 0, \tag{69}$$

so its trace weighted by $W_P\,W_P^\top \succeq 0$ is non-negative. The bias-correction term is a squared Euclidean norm and is also non-negative. Equality of either term to zero requires the corresponding factor to vanish, giving the stated equality condition. The ratio form follows by dividing the centered identity $\operatorname{tr}(W_P\Sigma_{PP}W_P^\top) - J_{\mathcal{D}}^\star = \operatorname{tr}(W_P\Sigma_{PS}\Sigma_{SS}^\dagger\Sigma_{SP}W_P^\top)$ by $\operatorname{tr}(W_P\Sigma_{PP}W_P^\top)$. □

Interpretation. Compensation strictly reduces the layer error whenever either (i) $W_P\mu_P \neq 0$ (pruned channels have non-zero mean and the bias correction in $\widehat{c}_\lambda$ matters, e.g., post-GELU MLPs), or (ii) the kept activations carry any linear signal about the pruned ones in the directions of W_P. The quantity $\rho_{W_P}^2$ is a multivariate coefficient of determination: it measures the fraction of the variance of $W_P x_P$ that is linearly explained by x_S. The bias and variance terms in (64) correspond exactly to the two parts of the closed-form solution in (50): $\widehat{c}_\lambda$ removes the mean term, and $\widehat{B}_\lambda$ captures the predictable variance. Both Σ_{PS} and Σ_{SS} are already computed during compensation (Algorithm 3), so $\rho_{W_P}^2$ is available at no additional cost; measured values are reported in Appendix F.1.

Proposition C.1.3 (calibration–test shift, sketch) *Let $\widehat{B}_\lambda, \widehat{c}_\lambda$ be the ridge estimators in* (50) *fitted from N i.i.d. calibration samples $\{x^{(i)}\}_{i=1}^N$ drawn from $\mathcal{D}$, and let $(B_t^\star, c_t^\star)$ be the population-optimal affine predictor under a deployment distribution $\mathcal{D}_t$. Define the test-time layer error*

$$\mathcal{E}_t(\widehat{B}_\lambda, \widehat{c}_\lambda) \;:=\; \mathbb{E}_{\mathcal{D}_t}\Big[\, \big\| W_P\,(x_P - \widehat{B}_\lambda x_S - \widehat{c}_\lambda)\big\|_2^2 \Big]. \tag{70}$$

Assume (A1) $\|x\|_2 \le R$ almost surely under both $\mathcal{D}$ and $\mathcal{D}_t$, and (A2) $\Sigma_{SS}^{(c)} + \lambda I \succeq \sigma_{\min} I$ for some $\sigma_{\min} > 0$. Then with probability at least $1-\delta$ over the calibration draw,

$$\mathcal{E}_t(\widehat{B}_\lambda, \widehat{c}_\lambda) \;\le\; 2\,J_{\mathcal{D}_t}^\star + C_1\,\|W_P\|_F^2\,\frac{R^4\,|S|}{\sigma_{\min}^2\,N}\,\log\frac{|S|}{\delta} + C_2\,\|W_P\|_F^2\,\Big(\big\|\Sigma^{(c)}-\Sigma^{(t)}\big\|_{\mathrm{op}} + \big\|\mu^{(c)}-\mu^{(t)}\big\|_2^2\Big), \tag{71}$$

for absolute constants $C_1, C_2 > 0$, where $J_{\mathcal{D}_t}^\star$ is the irreducible affine residual on $\mathcal{D}_t$ given by Proposition C.1.1 applied to $\mathcal{D}_t$.

Proof sketch. The argument follows three standard steps.

(i) Bias–variance split. Using the decomposition

$$x_P - \widehat{B}_\lambda x_S - \widehat{c}_\lambda = \big(x_P - B_t^\star x_S - c_t^\star\big) + \big(B_t^\star - \widehat{B}_\lambda\big)x_S + \big(c_t^\star - \widehat{c}_\lambda\big), \tag{72}$$

and the inequality $\|u+v\|_2^2 \le 2\|u\|_2^2 + 2\|v\|_2^2$ to obtain

$$\mathcal{E}_t(\widehat{B}_\lambda, \widehat{c}_\lambda) \le 2\,J_{\mathcal{D}_t}^\star + 2\,\|W_P\|_F^2\,\mathbb{E}_{\mathcal{D}_t}\big[\,\big\|(B_t^\star - \widehat{B}_\lambda)x_S + (c_t^\star - \widehat{c}_\lambda)\big\|_2^2\,\big]. \tag{73}$$

(ii) Estimation error on $\mathcal{D}$. Decompose $B_t^\star - \widehat{B}_\lambda = (B_t^\star - B_c^\star) + (B_c^\star - \widehat{B}_\lambda)$ where $B_c^\star$ is the population ridge solution on $\mathcal{D}$. Under (A1)–(A2), matrix Bernstein concentration applied to $\widehat{\Sigma}_{SS}^{(c)}$ and $\widehat{\Sigma}_{PS}^{(c)}$ [Tropp, 2015] yields

$$\big\|B_c^\star - \widehat{B}_\lambda\big\|_F^2 \;\le\; C_1'\,\frac{R^4\,|S|}{\sigma_{\min}^2\,N}\,\log\frac{|S|}{\delta}, \tag{74}$$

with probability at least $1-\delta$, and an analogous Hoeffding bound for $\mu^{(c)}$ controls $\|c_c^\star - \widehat{c}_\lambda\|_2^2$ at the same rate. The fast-rate variant of Hsu et al. [2014] gives the explicit constants under sub-Gaussian relaxations of (A1).

(iii) Distribution shift on $\mathcal{D}_t$. The remaining piece $B_t^\star - B_c^\star$ is bounded via standard matrix perturbation [e.g., Horn and Johnson, 2013, Cor. 6.3.8]: a C^1 perturbation of Σ_{SS} propagates through the (pseudo)inverse, yielding

$$\big\|B_t^\star - B_c^\star\big\|_F^2 \;\le\; C_2'\,\big\|\Sigma^{(c)} - \Sigma^{(t)}\big\|_{\mathrm{op}}^2/\sigma_{\min}^2, \tag{75}$$

and a triangle inequality on the means controls the bias drift.

Combining (i)–(iii) and absorbing constants into C_1, C_2 gives (71). □

Interpretation. The test-time error decomposes into three terms. The first, $2\,J_{\mathcal{D}_t}^\star$, is the irreducible bilinear residual on the deployment distribution and is the same Schur-complement quantity from Proposition C.1.1 applied to $\mathcal{D}_t$; it is the floor any distribution-conditional method must pay. The second term decays as $|S|/N$ and gives the calibration sample-complexity guidance: doubling $|S|$ (less aggressive pruning, larger kept set) doubles the calibration budget needed for the same estimation error, and the dependence on $\sigma_{\min}^{-2}$ explains why aggressive pruning that leaves a poorly conditioned Σ_{SS} inflates the bound. The third term grows linearly in the operator-norm gap between calibration and deployment covariances, providing a controlled (rather than catastrophic) sensitivity to mismatch and supporting the empirical robustness observed in our OPT–C4/WikiText-2 experiment (Section 4).

C.2 Attention QK

We now derive the analogous bounds for the attention compensation. Following the notation of Section 3.4, fix a layer and a head; let $Q, K \in \mathbb{R}^{n\times d_h'}$ denote per-sample query and key matrices

restricted to that head, and partition $Q = [\, Q_S \;\; Q_P \,]$, $K = [\, K_S \;\; K_P \,]$ along the head dimension into kept indices S and pruned indices P. The original logits decompose as

$$L \;=\; QK^\top \;=\; Q_S K_S^\top \;+\; \underbrace{Q_P K_P^\top}_{=:\,T}, \tag{76}$$

and the missing piece $T = Q_P K_P^\top$ is what compensation tries to recover through a small matrix $M \in \mathbb{R}^{d'_h \times d'_h}$.

Let $\mathcal{D} = \{x^{(b)}\}_{b=1}^N$ be the calibration set, and write $Q_{S,b}, K_{S,b}, Q_{P,b}, K_{P,b}, T_b$ for the per-sample matrices. The compensation objective from Eqs. (14) and (15) of the main text is the empirical ridge problem

$$\widehat{M}_\lambda \;:=\; \arg\min_M \; \sum_{b=1}^N \big\| T_b - Q_{S,b}\, M\, K_{S,b}^\top \big\|_F^2 + \lambda\, \|M\|_F^2. \tag{77}$$

Vectorizing M via $m := \mathrm{vec}(M)$ and using the Kronecker identity $\mathrm{vec}(Q_S\, M\, K_S^\top) = (K_S \otimes Q_S)\, \mathrm{vec}(M)$, define the empirical Gram tensor and cross-product

$$G \;:=\; \sum_{b=1}^N (K_{S,b}^\top K_{S,b}) \otimes (Q_{S,b}^\top Q_{S,b}) \;\in\; \mathbb{R}^{d'^2_h \times d'^2_h}, \tag{78}$$

$$h \;:=\; \sum_{b=1}^N \mathrm{vec}\Big((Q_{S,b}^\top Q_{P,b})(K_{P,b}^\top K_{S,b}) \Big) \;\in\; \mathbb{R}^{d'^2_h}. \tag{79}$$

The closed-form ridge solution is

$$\mathrm{vec}(\widehat{M}_\lambda) \;=\; (G + \lambda I)^\dagger\, h. \tag{80}$$

Proposition C.2.1 (exact attention layer error) *For any $\lambda \geq 0$, the ridge-compensated logit distortion satisfies*

$$J_{\mathcal{D}}(\widehat{M}_\lambda) \;:=\; \sum_{b=1}^N \big\| T_b - Q_{S,b}\, \widehat{M}_\lambda\, K_{S,b}^\top \big\|_F^2 \;=\; \sum_{b=1}^N \big\| T_b \big\|_F^2 \;-\; 2\, h^\top (G+\lambda I)^\dagger\, h \;+\; h^\top (G+\lambda I)^\dagger\, G\, (G+\lambda I)^\dagger\, h. \tag{81}$$

In the population limit $\lambda \to 0^+$ (or whenever G is invertible), this collapses to

$$J_{\mathcal{D}}^\star \;=\; \sum_{b=1}^N \big\| T_b \big\|_F^2 \;-\; h^\top G^\dagger\, h. \tag{82}$$

Proof. We proceed in four steps.

Step 1 (vectorize the objective). Let $m = \mathrm{vec}(M)$ and $t_b = \mathrm{vec}(T_b)$. The Kronecker identity $\mathrm{vec}(AMB^\top) = (B \otimes A)\, \mathrm{vec}(M)$ applied to $A = Q_{S,b}$, $B = K_{S,b}$ gives

$$\mathrm{vec}(Q_{S,b}\, M\, K_{S,b}^\top) \;=\; \Phi_b\, m, \qquad \Phi_b \;:=\; K_{S,b} \otimes Q_{S,b} \;\in\; \mathbb{R}^{n^2 \times d'^2_h}. \tag{83}$$

Hence the per-sample loss equals $\|T_b - Q_{S,b}\, M\, K_{S,b}^\top\|_F^2 = \|t_b - \Phi_b\, m\|_2^2$, and the empirical objective in (77) becomes

$$J_{\mathcal{D}}(M) + \lambda \|m\|_2^2 \;=\; \sum_{b=1}^N \|t_b - \Phi_b\, m\|_2^2 \;+\; \lambda\, \|m\|_2^2. \tag{84}$$

Step 2 (identify G and h). Expanding the squared norm,

$$\sum_{b=1}^N \|t_b - \Phi_b\, m\|_2^2 \;=\; \sum_{b=1}^N \|t_b\|_2^2 \;-\; 2\, m^\top \sum_{b=1}^N \Phi_b^\top t_b \;+\; m^\top \Big(\sum_{b=1}^N \Phi_b^\top \Phi_b \Big) m. \tag{85}$$

Two algebraic identities reduce this to the form involving G and h. First,

$$\Phi_b^\top \Phi_b \;=\; (K_{S,b} \otimes Q_{S,b})^\top (K_{S,b} \otimes Q_{S,b}) \;=\; (K_{S,b}^\top K_{S,b}) \otimes (Q_{S,b}^\top Q_{S,b}), \tag{86}$$

so $\sum_b \Phi_b^\top \Phi_b = G$. Second, using $(B \otimes A)^\top \mathrm{vec}(C) = \mathrm{vec}(A^\top C\, B)$ with $A = Q_{S,b}$, $B = K_{S,b}$, $C = T_b = Q_{P,b} K_{P,b}^\top$,

$$\Phi_b^\top t_b \;=\; \mathrm{vec}\big(Q_{S,b}^\top\, Q_{P,b} K_{P,b}^\top\, K_{S,b}\big) \;=\; \mathrm{vec}\big((Q_{S,b}^\top Q_{P,b})(K_{P,b}^\top K_{S,b})\big), \tag{87}$$

so $\sum_b \Phi_b^\top t_b = h$. Substituting into (84):

$$J_{\mathcal{D}}(M) + \lambda \|m\|_2^2 \;=\; \sum_{b=1}^{N} \|t_b\|_2^2 \;-\; 2\, m^\top h \;+\; m^\top (G + \lambda I)\, m. \tag{88}$$

Step 3 (substitute the ridge optimum). Plugging $\widehat{m}_\lambda = (G + \lambda I)^\dagger h$ into (88) and dropping the regularizer term to recover $J_{\mathcal{D}}$ alone,

$$J_{\mathcal{D}}(\widehat{M}_\lambda) = \sum_{b=1}^{N} \|t_b\|_2^2 \;-\; 2\, h^\top (G + \lambda I)^\dagger h \;+\; h^\top (G + \lambda I)^\dagger\, G\, (G + \lambda I)^\dagger\, h, \tag{89}$$

which is (81). Note that $\sum_b \|t_b\|_2^2 = \sum_b \|T_b\|_F^2$ by the Frobenius–ℓ_2 identity for $\mathrm{vec}(\cdot)$.

Step 4 (population limit). When $\lambda \to 0^+$ on the column space of G, $(G + \lambda I)^\dagger \to G^\dagger$, and the optimality condition $G\, G^\dagger\, h = h$ (since $h \in \mathrm{range}(G)$ by construction) gives

$$h^\top G^\dagger\, G\, G^\dagger\, h \;=\; h^\top G^\dagger\, h, \tag{90}$$

so the last two terms in (81) combine to $-h^\top G^\dagger h$, yielding (82). □

Interpretation. The minimum logit distortion equals the total squared norm of the missing logits $\sum_b \|T_b\|_F^2$ minus a non-negative quantity $h^\top G^\dagger h$ that quantifies how much of T_b is reconstructible from the kept Kronecker subspace $\mathrm{span}(K_{S,b} \otimes Q_{S,b})$. The matrix G and vector h are exactly the LHS and RHS of the calibration-summed normal equation (44) (Appendix B.2) and are computed during compensation, so this exact error is available at no additional cost.

Proposition C.2.2 (attention compensation gain) *Define the uncompensated logit distortion*

$$J_{\mathrm{uncomp}}^{\mathrm{attn}} \;:=\; \sum_{b=1}^{N} \big\| Q_P\, K_P^\top \big\|_F^2 \big|_{x = x^{(b)}} \;=\; \sum_{b=1}^{N} \|T_b\|_F^2, \tag{91}$$

obtained by setting $M = 0$ in (77). Then in the population limit ($\lambda \to 0^+$) the compensation gain is

$$J_{\mathrm{uncomp}}^{\mathrm{attn}} \;-\; J_{\mathcal{D}}^{\star} \;=\; h^\top G^\dagger\, h \;\geq\; 0, \tag{92}$$

with equality (zero gain) iff $h = 0$, i.e., the kept Q/K subspace is orthogonal to the missing logits in the bilinear sense $Q_{S,b}^\top Q_{P,b} K_{P,b}^\top K_{S,b} = 0$ for almost every b. Equivalently, defining the bilinear coefficient of determination

$$\rho_{\mathrm{attn}}^2 \;:=\; \frac{h^\top G^\dagger\, h}{\sum_{b=1}^{N} \|T_b\|_F^2} \;\in\; [0, 1], \tag{93}$$

the ratio satisfies $J_{\mathcal{D}}^{\star} / J_{\mathrm{uncomp}}^{\mathrm{attn}} = 1 - \rho_{\mathrm{attn}}^2$.

Proof. We proceed in two steps.

Step 1 (uncompensated error). Setting $m = 0$ in (88) gives

$$J_{\mathcal{D}}(M{=}0) \;=\; \sum_{b=1}^{N} \|t_b\|_2^2 \;=\; \sum_{b=1}^{N} \|T_b\|_F^2 \;=\; J_{\mathrm{uncomp}}^{\mathrm{attn}}. \tag{94}$$

Step 2 (subtract the compensated optimum). By Proposition C.2.1 in the population limit (82), $J_{\mathcal{D}}^{\star} = \sum_b \|T_b\|_F^2 - h^\top G^\dagger h$. Subtracting,

$$J_{\mathrm{uncomp}}^{\mathrm{attn}} - J_{\mathcal{D}}^{\star} \;=\; h^\top G^\dagger\, h. \tag{95}$$

The matrix G is positive semidefinite as a sum of Kronecker products of PSD matrices,

$$G \;=\; \sum_{b=1}^{N} (K_{S,b}^\top K_{S,b}) \otimes (Q_{S,b}^\top Q_{S,b}) \;\succeq\; 0, \tag{96}$$

so $G^\dagger$ is also PSD on $\mathrm{range}(G)$, and $h \in \mathrm{range}(G)$ by construction (each summand of h lies in the same column span as the corresponding summand of G). Hence $h^\top G^\dagger h \geq 0$, with equality iff $h = 0$. Dividing by $J_{\mathrm{uncomp}}^{\mathrm{attn}}$ yields the ratio form (93). □

Interpretation. ρ^2_{attn} is the attention analogue of the regression R^2 in Proposition C.1.2: it measures the fraction of the squared Frobenius norm of the missing logits $T_b = Q_{P,b}K_{P,b}^\top$ that lies in the bilinear span of $Q_{S,b}\, M\, K_{S,b}^\top$ over M. Compensation strictly improves on uncompensated pruning whenever $\rho^2_{\mathrm{attn}} > 0$, i.e., whenever the kept Q/K subspaces carry any bilinear signal about the removed ones. Both G and h are formed during compensation, so per-layer or per-head ρ^2_{attn} values can be reported as a diagnostic without any additional computation.

Proposition C.2.3 (attention calibration–test shift, sketch) *Let $\widehat{M}_\lambda$ be the ridge solution in* (80) *fitted from N i.i.d. calibration samples drawn from $\mathcal{D}$. Let $M_t^\star$ be the population-optimal compensator on a deployment distribution $\mathcal{D}_t$, and define the test-time distortion*

$$\mathcal{E}_t(\widehat{M}_\lambda) \;:=\; \mathbb{E}_{\mathcal{D}_t}\big[\, \|Q_P K_P^\top - Q_S\, \widehat{M}_\lambda\, K_S^\top\|_F^2 \,\big]. \tag{97}$$

Assume (B1) $\|q_j\|_2,\ \|k_j\|_2 \le R$ almost surely under both $\mathcal{D}$ and $\mathcal{D}_t$ for every column index j, and (B2) $\mathbb{E}_{\mathcal{D}}[G/N] + \lambda I \succeq \sigma_{\min} I$ for some $\sigma_{\min} > 0$, where G is normalized by sample count. Then with probability at least $1-\delta$ over the calibration draw,

$$\mathcal{E}_t(\widehat{M}_\lambda) \;\le\; 2\, J^\star_{\mathcal{D}_t} \;+\; C_3\, \frac{R^4\, d_h'^2}{\sigma^2_{\min} N}\, \log\frac{d'_h}{\delta} \;+\; C_4 \left(\big\| G^{(c)}/N - G^{(t)} \big\|_{\mathrm{op}} + \big\| h^{(c)}/N - h^{(t)} \big\|_2^2 \right), \tag{98}$$

for absolute constants $C_3, C_4 > 0$, where $J^\star_{\mathcal{D}_t}$ is the irreducible bilinear residual on $\mathcal{D}_t$ given by Proposition C.2.1 applied to $\mathcal{D}_t$, and $G^{(t)}$, $h^{(t)}$ are the population versions on $\mathcal{D}_t$.

Proof sketch. Three standard steps, in direct analogy to Proposition C.1.3.

(i) Bias–variance split. Add and subtract the test-optimal compensator $M_t^\star$ inside the residual:

$$Q_P K_P^\top - Q_S\, \widehat{M}_\lambda\, K_S^\top \;=\; \big(Q_P K_P^\top - Q_S\, M_t^\star\, K_S^\top\big) \;+\; Q_S\, (M_t^\star - \widehat{M}_\lambda)\, K_S^\top. \tag{99}$$

Applying $\|A+B\|_F^2 \le 2\|A\|_F^2 + 2\|B\|_F^2$ and taking expectations under $\mathcal{D}_t$ gives

$$\mathcal{E}_t(\widehat{M}_\lambda) \le 2\, J^\star_{\mathcal{D}_t} + 2\, \mathbb{E}_{\mathcal{D}_t}\big[\, \|Q_S\, (M_t^\star - \widehat{M}_\lambda)\, K_S^\top\|_F^2 \,\big]. \tag{100}$$

(ii) Estimation error on $\mathcal{D}$. Decompose $M_t^\star - \widehat{M}_\lambda = (M_t^\star - M_c^\star) + (M_c^\star - \widehat{M}_\lambda)$ where $M_c^\star$ is the population ridge solution on $\mathcal{D}$. Under (B1)–(B2), matrix Bernstein concentration applied to the empirical versions of G and h [Tropp, 2015] yields

$$\big\|\mathrm{vec}(M_c^\star - \widehat{M}_\lambda)\big\|_2^2 \;\le\; C_3'\, \frac{R^4\, d_h'^2}{\sigma^2_{\min} N}\, \log\frac{d'_h}{\delta}, \tag{101}$$

with probability at least $1-\delta$. The dimension of $\mathrm{vec}(M)$ is $d_h'^2$, which appears in both the dimension factor of Bernstein and the log term. The fast-rate analysis of Hsu et al. [2014], transferred to the bilinear ridge setting, gives the explicit constants under sub-Gaussian relaxations of (B1).

(iii) Distribution shift. Bound $M_t^\star - M_c^\star$ via pseudoinverse perturbation [e.g., Stewart, 1977]: a C^1 perturbation of G propagates through $G^\dagger$, giving

$$\big\|\mathrm{vec}(M_t^\star - M_c^\star)\big\|_2^2 \;\le\; C_4'\, \frac{\|G^{(c)}/N - G^{(t)}\|_{\mathrm{op}}^2}{\sigma^2_{\min}} \;+\; C_4''\, \frac{\|h^{(c)}/N - h^{(t)}\|_2^2}{\sigma^2_{\min}}. \tag{102}$$

Combining (i)–(iii), passing the inner expectation through Q_S, K_S using boundedness from (B1), and absorbing constants into C_3, C_4 gives (98). □

Interpretation. The decomposition mirrors Proposition C.1.3. The first term $2\, J^\star_{\mathcal{D}_t}$ is the irreducible bilinear residual on $\mathcal{D}_t$ (Proposition C.2.1 applied to deployment) and is the floor any layer-local method must pay. The second term decays as $d_h'^2/N$ and gives sample-complexity guidance specific to attention: the per-head compensator has only $d_h'^2$ free parameters (e.g., $32^2 = 1024$ or $64^2 = 4096$), so the calibration budget required for stable estimation is much smaller than for MLP compensation, where $|S|$ can reach the thousands. This explains empirically why attention-only pruning is more stable than MLP-only pruning at small calibration sizes (Table 3). The third term grows linearly in the operator-norm gap of the empirical Gram tensors and provides a controlled sensitivity to calibration–deployment mismatch.

D Detailed Algorithms

This appendix provides detailed pseudocode for the components of CORP referenced in the main text. Algorithm 1 summarizes the full pipeline, and the following algorithms detail each step.

D.1 MLP Channel Ranking

Algorithm 2 RankMLP: Hidden-Channel Ranking for MLP Pruning

Input: Hidden activations $x \in \mathbb{R}^d$, sparsity s_{mlp}
Output: Kept indices S, pruned indices P
for $j = 1$ **to** d **do**
  Compute score $r_j \leftarrow \mathbb{E}[x_j^2]$ {activation-based}
end for
Select top $(1 - s_{\mathrm{mlp}})d$ indices by r_j as S
$P \leftarrow \{1, \ldots, d\} \setminus S$
return (S, P)

Algorithm 2 specifies the hidden-channel ranking procedure used for MLP pruning. The ranker scores channels using activation statistics computed on the calibration set and selects channels to preserve based on the target sparsity.

D.2 MLP Affine Compensation

Algorithm 3 CompensateMLP: Affine Compensation for Pruned MLP Channels

Input: Layer $y = Wx + b$, activations x, indices (S, P), ridge λ
Output: Compensated parameters $(\widehat{W}_S, \widehat{b})$
Form calibration matrices X_S, X_P from x_S, x_P
Compute $\mu_S = E[x_S]$, $\mu_P = E[x_P]$
Compute $\Sigma_{SS} = E[(x_S - \mu_S)(x_S - \mu_S)^\top]$
Compute $\Sigma_{PS} = E[(x_P - \mu_P)(x_S - \mu_S)^\top]$
$B \leftarrow \Sigma_{PS}(\Sigma_{SS} + \lambda I)^{-1}$
$c \leftarrow \mu_P - B\mu_S$
$\widehat{W}_S \leftarrow W_S + W_P B$
$\widehat{b} \leftarrow b + W_P c$
return $(\widehat{W}_S, \widehat{b})$

Algorithm 3 details the closed-form affine compensation used to correct the bias introduced by MLP channel pruning. The procedure estimates second-order statistics from calibration activations and folds the compensation directly into the output projection weights.

D.3 Attention Dimension Ranking

Algorithm 4 RankAttn: Logit-Energy Ranking for Attention Pruning

Input: Queries Q, keys K, sparsity s_{attn}
Output: Kept indices S, pruned indices P
for $j = 1$ **to** d_h **do**
  $s_j \leftarrow \mathbb{E}[\|q_j\|_2^2 \|k_j\|_2^2]$
end for
Select top $(1 - s_{\mathrm{attn}})d_h$ indices by s_j as S
$P \leftarrow \{1, \ldots, d_h\} \setminus S$
return (S, P)

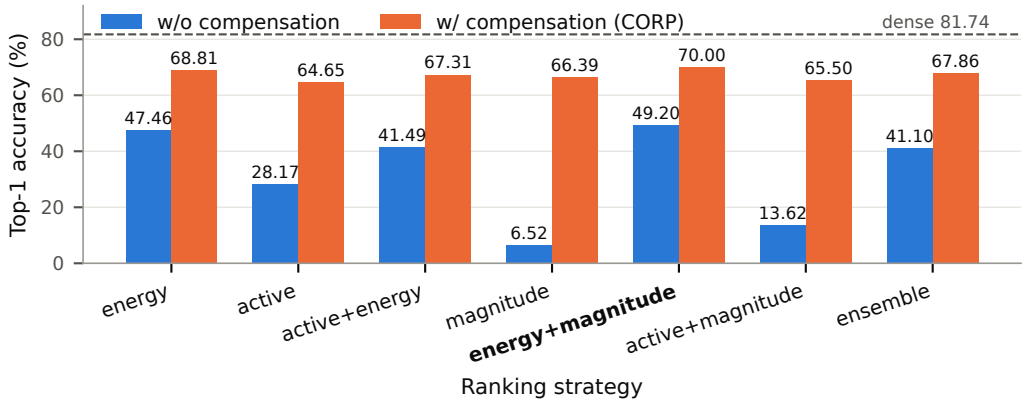


Figure 5: Comparison of ranking strategies with and without compensation. Activation-based and magnitude-based criteria show varied performance, while combined strategies achieve higher accuracy. Compensation consistently improves results across all ranking methods, highlighting that representation recovery is more critical than the choice of ranking.

Algorithm 4 presents the logit-energy based ranking for attention projection pruning. Each head dimension is scored by its expected contribution to the attention logits, allowing structured pruning without modifying the attention layout.

D.4 Attention Logit Compensation

Algorithm 5 CompensateAttn: Logit-Space Compensation for Attention Pruning

Input: Projections (W_Q, W_K), queries Q, keys K, indices (S, P), ridge λ
Output: Compensated projections $(\hat{W}_{Q,S}, \hat{W}_{K,S})$
Split $Q = [Q_S\ Q_P]$, $K = [K_S\ K_P]$
Initialize $G \leftarrow 0$, $h \leftarrow 0$
for each calibration sample b **do**
  $G \leftarrow G + (K_{S,b}^\top K_{S,b}) \otimes (Q_{S,b}^\top Q_{S,b})$
  $h \leftarrow h + \mathrm{vec}\Big((Q_{S,b}^\top Q_{P,b})(K_{P,b}^\top K_{S,b})\Big)$
end for
Solve $(G + \lambda I)\,\mathrm{vec}(M) = h$
$M \leftarrow \mathrm{mat}(\mathrm{vec}(M))$
Compute SVD $I + M = U\Sigma V^\top$
$\hat{W}_{Q,S} \leftarrow W_{Q,S} U \Sigma^{1/2}$
$\hat{W}_{K,S} \leftarrow W_{K,S} V \Sigma^{1/2}$
return $(\hat{W}_{Q,S}, \hat{W}_{K,S})$

Algorithm 5 describes the logit-space compensation scheme for attention pruning. The method recovers missing logit contributions by solving a small Sylvester equation and folding the solution into the retained query and key projections.

E Ranking ablation

We compare activation energy, magnitude-based, active probability and combined ranking strategies from Sec. 3.3, with and w/o compensation on DeiT-Base at 50% joint sparsity.

The active ranking policy scores each channel by its activation frequency, $P(|x| > \epsilon)$, estimated over the calibration set. Channels that activate less frequently are assigned higher pruning priority. The intuition is that rarely activated channels contribute little to downstream representations, so removing

them introduces minimal error regardless of weight magnitude. The combination of activation energy, magnitude-based strategy achieves the best accuracy and is used by default. The result is in Figure 5

F Additional Analyses

F.1 Empirical Recoverability

Propositions C.1.2 and C.2.2 define $\rho^2_{W_P}$ and ρ^2_{attn} as the fraction of the pruned contribution's energy that is linearly recoverable from the retained components. We measure both on DeiT-Large at 50% joint MLP and attention sparsity with the default energy–magnitude ranking and $\lambda = 10^{-3}$. The calibration and evaluation sets are disjoint random samples of 4000 ImageNet training images each, both drawn from all 1000 classes. For a unit u (an MLP layer or an attention head), let E_u denote the energy of the removed contribution and R_u the residual energy after compensation, so that $\rho^2_u = 1 - R_u/E_u$ is the fraction of that unit's removed energy recovered by linear compensation. We report the *pooled* value

$$\rho^2_{\mathrm{pool}} \;=\; 1 - \frac{\sum_u R_u}{\sum_u E_u} \;=\; \frac{\sum_u E_u\, \rho^2_u}{\sum_u E_u}, \tag{103}$$

which is the fraction of the *total* pruned energy recovered and therefore the quantity that corresponds to the representation distortion removed by CORP, together with the unweighted mean, standard deviation, and range over units, which show how uniformly compensation works. Three protocols are used:

- **Calibration**: fit and evaluate the compensator on the calibration set (in-sample recoverability).
- **Evaluation**: fit and evaluate on the evaluation set (an oracle upper bound that is not available to CORP).
- **Transfer**: fit the compensator once on the calibration set, freeze it, and evaluate on the held-out evaluation set. This matches CORP's deployment; the gap between evaluation and transfer is the cost of fitting only once.

Table 10 summarizes the results and Figure 6 shows the per-layer and per-head distributions. Fitting the compensator once costs almost nothing relative to the oracle: the pooled oracle-to-transfer gap is 0.0007 for attention ($0.7775 \to 0.7768$) and 0.0089 for the MLP ($0.4825 \to 0.4736$). The pooled and unweighted statistics differ because the removed attention-logit energy is highly concentrated: the top 10 of 384 heads account for 56.4% of the removed energy and have a mean ρ^2 of 0.980, so the pooled value is dominated by the most influential heads. Quoting only the pooled value would overstate how uniformly compensation works, while quoting only the unweighted mean would understate how much representation distortion it removes. Recoverability also decreases with depth for both module types (Figure 6), and the largest transfer gaps occur in the deeper MLP blocks.

Protocol	Module	Pooled ρ^2	Mean $\pm$ std.	Min	Max
Calibration	MLP (24 layers)	0.4893	0.3944 ± 0.1707	0.1126	0.7816
Evaluation (oracle)	MLP	0.4825	0.3909 ± 0.1698	0.1133	0.7766
Transfer (CORP)	MLP	**0.4736**	0.3755 ± 0.1789	0.0619	0.7735
Calibration	Attention (384 heads)	0.7825	0.4018 ± 0.2336	0.1045	0.9995
Evaluation (oracle)	Attention	0.7775	0.3993 ± 0.2323	0.1018	0.9995
Transfer (CORP)	Attention	**0.7768**	0.3970 ± 0.2335	0.1008	0.9995

Table 10: Recoverability ρ^2 on DeiT-Large (50% joint sparsity) under the three protocols. Pooled values are energy weighted; mean, standard deviation, minimum, and maximum are unweighted over units.

Content dependence. Because M is fitted once from aggregate statistics while attention patterns are content dependent, we repeat the transfer protocol under a stricter split: the compensator is fitted on images from 500 ImageNet classes and evaluated on the remaining 500 classes, with 2000 images per side, so the fitted and evaluated inputs share no object category. A same-distribution control with

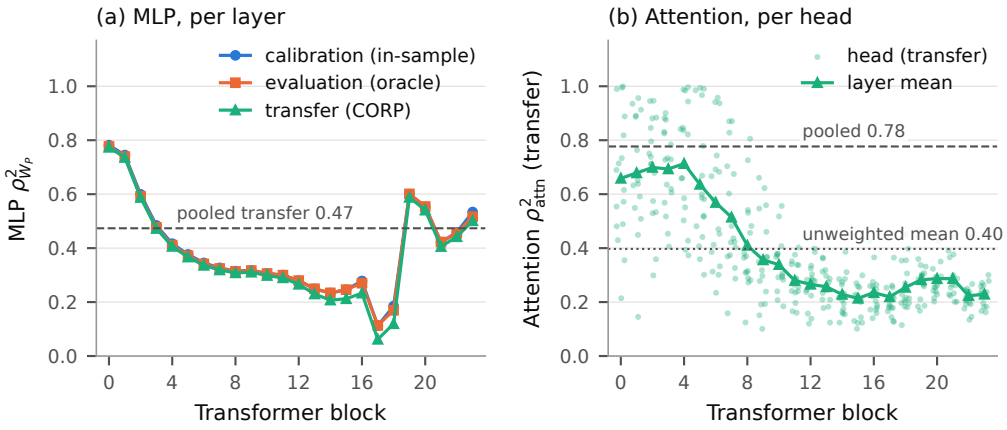


Figure 6: Layer-wise and head-wise recoverability on DeiT-Large (50% joint sparsity). **(a)** MLP $\rho^2_{W_P}$ per block under the three protocols; the three curves nearly coincide, so fitting once on the calibration set loses little. **(b)** Transfer ρ^2_{attn} of each of the 384 heads (dots) and the per-block mean (line); the gap between the energy-weighted pooled value and the unweighted mean reflects the concentration of removed logit energy in a few highly recoverable heads.

the same sample size isolates the effect of the smaller sample. Table 11 shows that the attention gap under the class-disjoint split (0.0016) is within 0.0004 of the control (0.0012), so almost all of the degradation comes from finite-sample estimation rather than content shift; MLP compensation is more sensitive (0.0233 vs. 0.0179). These results support transfer to unseen inputs and unseen object categories from the same deployment distribution; they do not establish robustness under arbitrary distribution shift.

Split	Module	Oracle (eval)	Transfer	Gap	Transfer mean $\pm$ std.
Class-disjoint (500/500 classes)	MLP	0.4771	0.4538	0.0233	0.3437 ± 0.1810
Class-disjoint (500/500 classes)	Attention	0.7795	0.7779	0.0016	0.3942 ± 0.2352
Same-distribution control	MLP	0.4866	0.4687	0.0179	0.3674 ± 0.1838
Same-distribution control	Attention	0.7779	0.7767	0.0012	0.3973 ± 0.2339

Table 11: Pooled ρ^2 under a class-disjoint calibration/evaluation split and a same-distribution control of matched sample size (2000 images per side). Gap is oracle minus transfer.

F.2 Ridge Coefficient Sensitivity

We use one fixed value, $\lambda = 10^{-3}$, for every MLP and attention solve in all reported experiments; λ is not tuned per model, layer, head, or sparsity ratio, and the ridge term only regularizes each covariance or normal-equation solve. Table 12 reports Top-1 accuracy on DeiT-Large under 50%, 60%, and 70% joint sparsity for $\lambda \in \{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$. Over 10^{-5}–10^{-3} the standard deviation stays below 0.06 points at every sparsity level, including 70%, where the retained covariance is expected to be less well conditioned. The larger value 10^{-2} reduces accuracy slightly (by 0.66 points at 70%), which indicates mild over-regularization rather than numerical instability. CORP therefore does not require setting-specific tuning of λ; we selected 10^{-3} as a single conservative default.

Joint sparsity	$\lambda = 10^{-5}$	10^{-4}	10^{-3} (default)	10^{-2}	Mean $\pm$ std. (10^{-5}–10^{-3})
50%	80.34	80.33	80.30	80.06	80.32 ± 0.02
60%	75.95	75.96	75.92	75.57	75.94 ± 0.02
70%	64.49	64.49	64.40	63.74	64.46 ± 0.05

Table 12: Top-1 accuracy of DeiT-Large under joint sparsity as a function of the ridge coefficient λ.

G Efficiency

Model	Sparsity	Top-1 (%)	Param (M)	FLOPs (G)	Lat (ms)	TP (fps)	Param↓ (%)	FLOPs↓ (%)	TP↑ (×)
DeiT-T	0.0	72.02	5.7	1.4	3.79	3514	0.0	0.0	1.00
	0.1	71.26	5.3	1.2	4.95	3247	7.6	19.0	0.92
	0.2	69.15	4.8	1.1	5.01	3453	15.3	25.6	0.98
	0.3	64.46	4.4	1.0	4.84	3554	23.2	32.4	1.01
	0.4	55.73	4.0	0.9	4.89	3699	30.9	39.0	1.05
	0.5	41.36	3.5	0.8	4.91	4048	38.8	45.9	1.15
	0.6	21.02	3.1	0.7	5.06	4136	46.4	52.4	1.18
	0.7	2.33	2.6	0.6	5.03	4434	54.1	59.0	1.26
DeiT-S	0.0	79.72	22.1	5.0	3.71	1370	0.0	0.0	1.00
	0.1	78.48	20.3	4.2	4.96	1334	7.9	14.5	0.97
	0.2	76.78	18.6	3.9	5.25	1440	15.9	21.7	1.05
	0.3	74.53	16.8	3.5	4.99	1499	24.0	29.3	1.09
	0.4	70.84	15.0	3.2	4.96	1621	32.0	36.5	1.18
	0.5	58.37	13.2	2.8	4.91	1831	40.2	44.1	1.34
	0.6	34.54	11.4	2.4	5.04	1917	48.1	51.4	1.40
	0.7	15.64	9.7	2.1	5.17	2108	56.1	58.6	1.54
DeiT-B	0.0	81.74	86.6	18.3	4.28	418	0.0	0.0	1.00
	0.1	80.26	79.6	16.2	4.94	431	8.1	11.6	1.03
	0.2	79.14	72.6	14.8	4.90	463	16.2	19.3	1.11
	0.3	76.98	65.4	13.3	5.29	502	24.5	27.3	1.20
	0.4	74.23	58.4	11.9	4.91	549	32.6	35.0	1.31
	0.5	70.00	51.2	10.4	4.91	630	40.9	43.0	1.51
	0.6	63.33	44.2	9.0	4.88	685	49.0	50.7	1.64
	0.7	49.78	37.2	7.6	4.90	766	57.1	58.4	1.83
DeiT-L	0.0	84.58	304.4	63.5	12.96	121	0.0	0.0	1.00
	0.1	84.57	279.5	56.6	12.43	129	8.2	10.8	1.06
	0.2	84.35	254.7	51.6	11.41	140	16.3	18.7	1.15
	0.3	83.61	229.0	46.5	10.62	151	24.8	26.8	1.25
	0.4	82.75	204.1	41.5	10.66	166	33.0	34.7	1.37
	0.5	80.30	178.5	36.3	10.61	187	41.4	42.8	1.54
	0.6	75.92	153.6	31.4	10.57	207	49.5	50.6	1.71
	0.7	64.40	128.8	26.4	10.58	238	57.7	58.5	1.96
DeiT-H	0.0	84.97	632.1	172.8	30.84	43	0.0	0.0	1.00
	0.1	84.96	579.7	153.7	27.70	48	8.3	11.1	1.12
	0.2	84.91	527.2	139.9	24.82	52	16.6	19.0	1.21
	0.3	84.66	474.8	126.2	22.73	57	24.9	27.0	1.32
	0.4	84.23	422.3	112.4	20.70	63	33.2	35.0	1.46
	0.5	83.27	369.9	98.7	19.16	71	41.5	42.9	1.64
	0.6	80.98	317.4	84.9	16.83	78	49.8	50.9	1.82
	0.7	75.38	265.0	71.2	14.71	92	58.1	58.8	2.13

Table 13: Accuracy and efficiency trade-offs of CORP across model scales. We report Top-1 accuracy, parameters (M), FLOPs (G), latency (ms), and throughput (fps). FLOPs and throughput are measured on a single RTX 3090 GPU with batch sizes 256/128/64/32/16 for Tiny/Small/Base/Large/Huge. Latency uses batch size 1.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the main claims: one-shot structured pruning with closed-form compensation, no labels, gradients, or fine-tuning, and empirical validation on DeiT, OPT, and DINOv2.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses limitations, including independent module-wise compensation, uniform layer sparsity, calibration-test mismatch, very high sparsity, and lack of global task-loss optimality.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The paper states the local representation-recovery assumptions and provides derivations for MLP affine compensation, attention logit compensation, and calibration-shift bounds in the appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper specifies datasets, models, calibration sizes, sparsity settings, metrics, hardware, and algorithms needed to reproduce the main experiments.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The paper uses public datasets and pretrained models. An anonymized code repository is provided at submission time, including instructions for environment setup, data preparation, pruning, compensation, and reproducing the main experiments and efficiency evaluations.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper reports calibration and evaluation datasets, model families, sparsity settings, metrics, batch sizes for efficiency evaluation, and hardware details. The ridge coefficient is fixed to $\lambda = 10^{-3}$ in all experiments, and its sensitivity is reported in Appendix F.2.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: For the main results (Table 2: all five models and all three pruning targets at 50% sparsity) we report mean $\pm$ sample standard deviation over five independently sampled

calibration sets; the variability captured is calibration-set sampling, and the standard deviation is computed with the sample (unbiased) formula. Other entries are single deterministic runs, as stated in the table captions.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper states that experiments run on a single 24GB NVIDIA RTX 3090 GPU and reports runtime breakdowns across model scales.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work uses public datasets and pretrained models for model compression research and does not involve human subjects or sensitive data collection.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The method may reduce inference cost, energy use, and deployment barriers. Potential risks include enabling cheaper deployment of harmful or biased models, so compressed models should inherit the same safety checks as their dense counterparts.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release new pretrained models, datasets, or high-risk generative systems.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper cites the datasets and pretrained model families used in experiments, including ImageNet, C4, WikiText-2, NYUv2, ADE20K, DeiT, OPT, and DINOv2.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The paper does not introduce a new dataset, benchmark, or pretrained model asset.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The work does not involve crowdsourcing or human-subject experiments.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The work does not involve human-subject research.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are not used as a core methodological component of the proposed pruning method.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.