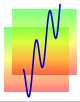
CrossSpectra: Exploiting Cross-Layer Smoothness for Parameter-Efficient Fine-Tuning



Yifei Zhang[1,2]*, **Hao Zhu**[4*], **Junhao Dong**[2], **Haoran Shi**[2], **Ziqiao Meng**[3], **Piotr Koniusz**[4,5,6 †] and **Han Yu**[2 †]
[1]Northwestern Polytechnical University;
[2]Nanyang Technological University; [3]National University of Singapore;
[4]Data61♥CSIRO; [5]University of New South Wales; [6]Australian National University
yifeiacc@gmail.com

Abstract

Parameter-efficient fine-tuning (PEFT) is essential for adapting large foundation models without excessive storage cost. However, current approaches such as LoRA treat each layer's adaptation independently, overlooking correlations across layers. This independence causes the number of trainable parameters to grow linearly with model depth. We provide theoretical and empirical evidence that skip connections in transformers create smooth gradient propagation across layers. This smoothness leads to weight adaptations that concentrate most of their energy in low-frequency spectral components, especially along the layer dimension. Empirical analysis confirms this effect, showing that most of adaptation energy lies in low frequencies. Building on this insight, we propose CrossSpectra, which parameterizes all attention-weight adaptations $(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V})$ across layers as a single 3D tensor and represents them with sparse spectral coefficients (κ_1, κ_2). Using κ_1 non-zero coefficients within each layer's frequency space and truncating to κ_2 frequencies across layers, CrossSpectra requires $\mathcal{O}(\kappa_1\kappa_2)$ parameters instead of LoRA's $\mathcal{O}(Lrd)$, where L is the number of layers and r is the rank. Across natural language and vision benchmarks, CrossSpectra matches or surpasses baseline performance while using fewer parameters than LoRA, achieving only 0.36% of LoRA's parameter count when fine-tuning LLaMA-7B on instruction-following tasks. These results show that exploiting the **architectural smoothness of transformers** through spectral analysis yields major efficiency gains in PEFT.

1 Introduction

Large Language models (LLMs) have demonstrated exceptional capabilities across domains ranging from natural language processing [Liu et al., 2019, He et al., 2020, Radford et al., 2019, Brown et al., 2020] to computer vision [Liu et al., 2023b,a, Singh et al., 2022, Zhang et al., 2024c] to multi-modal problems [Radford et al., 2021, Dong et al., 2025a,b,c,d, Zhang et al., 2025a,b]) and other domains [Fan et al., 2025, Zhang et al., 2024a, Guo et al., 2024]. However, with model parameters now reaching hundreds of billions or even trillions, adapting these models to specific downstream tasks through conventional fine-tuning has become increasingly impractical. Each customized model typically requires storing as many parameters as the original model [Qiu et al., 2020, Raffel et al., 2020a], leading to substantial storage and deployment challenges as customization needs expand.

Parameter-efficient fine-tuning (PEFT) methods have emerged as promising solutions to this challenge Yang et al. [2024]. These approaches adapt pre-trained models using only a small subset of

*Equal Contribution.
†Corresponding Authors.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

trainable parameters, significantly reducing storage requirements while maintaining performance. Among these methods, Low-Rank Adaptation (LoRA) Hu et al. [2021a] and its variants [Liu et al., 2024, Meng et al., 2024, Wang et al., 2024b, Kalajdzievski, 2023, Si et al., 2024, Zhong et al., 2024, Wang et al., 2024a, Ni et al., 2024]) have gained widespread adoption by representing weight changes through low-rank matrices, achieving impressive results with a fraction of the parameters required for full fine-tuning. Despite these advances, current PEFT methods face a fundamental limitation: they treat each layer's adaptation independently, overlooking potential structural relationships across layers. This independence assumption results in parameter counts that scale linearly with model depth, limiting efficiency gains for increasingly deeper architectures.

A fundamental insight from transformer architecture analysis illuminates a path toward more efficient adaptation. Skip connections, widely used in architectures including ResNets [He et al., 2016] and Transformers [Vaswani et al., 2017], facilitate stable and smooth gradient propagation through depth. This architectural property, combined with the natural spectral bias of neural networks [Rahaman et al., 2019], suggests that weight adaptations during fine-tuning should exhibit exploitable smoothness patterns, particularly across the layer dimension. Empirical analysis confirms this intuition. Figure 1 demonstrates that attention weight adaptations exhibit dramatic spectral concentration across layers, with nearly 70% of adaptation energy concentrated in low-frequency components. This observation motivates a spectral approach to parameter-efficient fine-tuning.

We formalize this intuition through theoretical analysis showing that attention weight adaptations in transformers naturally concentrate in low-frequency components. Our key theoretical findings are: (1) skip connections create Lipschitz-continuous gradient fields across layers, ensuring similar gradients in adjacent layers; (2) these smooth gradients accumulate into smooth weight adaptations during fine-tuning; and (3) the resulting spectral structure shows strongest decay in the cross-layer dimension, enabling aggressive frequency truncation. In this paper, we propose `CrossSpectra`, a novel PEFT method that exploits these cross-layer spectral properties. Instead of parameterizing each layer's adaptations independently, we construct a unified 3D tensor representation of all attention (Query, Key, Value) weight changes and decompose it in the frequency domain. By working directly with spectral coefficients and leveraging 3D inverse FFT for efficient computation, we achieve parameter reductions that scale sub-linearly with model dimensions. Our key contributions are:

i. We establish a theoretical framework connecting skip connection-induced gradient smoothness to spectral properties of weight adaptations, providing a principled foundation for cross-layer parameter sharing in attention mechanisms.

ii. We introduce `CrossSpectra`, a unified tensor formulation that represents all QKV adaptations across layers using sparse spectral decomposition with κ_1 coefficients per layer and κ_2 cross-layer frequencies, achieving $\mathcal{O}(\kappa_1\kappa_2)$ parameter complexity.

iii. We demonstrate that our approach achieves 275× parameter reduction compared to LoRA and 5,250× compared to full fine-tuning, requiring only 8 KB of memory versus LoRA's 2.2 MB for typical transformer configurations.

iv. Through extensive experiments across natural language understanding, instruction tuning, and image classification tasks, we show that `CrossSpectra` matches or exceeds baseline performance while using a fraction of the parameters.

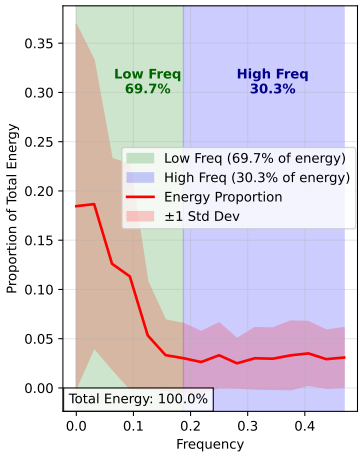


Figure 1: Distribution of spectral energy. Most of adaptation energy concentrates in low frequencies, confirming that skip connections induce smooth, low-frequency dominated weight changes across layer.

2 Related Work

Parameter-efficient Fine-tuning (PEFT) Methods. As foundation models have grown in size and computational requirements, efficiently adapting them has become essential. PEFT methods fall into two primary categories: non-weight-based [He et al., 2021, Rebuffi et al., 2017, Lester et al.,

2021, Gao et al., 2020, Li and Liang, 2021] and weight-based approaches [Zhu et al., 2025, Hu et al., 2021b, Zhang et al., 2025c, 2024b]. Weight-based methods directly learn modifications to pre-trained weights. LoRA [Hu et al., 2021b] represents weight changes through low-rank matrix decomposition. LoRA has gained widespread adoption due to its simplicity, effectiveness, and ability to merge adapted weights during inference to avoid latency increases. Several extensions have been proposed, including AdaLoRA [Zhang et al., 2023], which adaptively allocates parameter budgets across weight matrices, and FourierFT [Gao et al., 2024] and BiLoRA [Zhu et al., 2025] learn in frequency domain. PACE [Ni et al., 2024] uses noise modulation for generalization guarantees. Despite these advances, all existing methods treat each layer's adaptation independently, failing to exploit potential structural relationships across layers. This independent treatment results in parameter counts that scale linearly with model depth, limiting efficiency gains for increasingly deeper architectures.

Skip Connections in Deep Learning. Skip connections, first popularized in ResNet [He et al., 2016] and now fundamental to modern architectures like Transformers [Vaswani et al., 2017], allow information to bypass one or more layers by adding the input of a layer block to its output. Their empirical success in enabling training of very deep networks has been well-documented, but the theoretical understanding of their properties continues to evolve. A pivotal insight came from Chen et al. [2018], who established connections between networks with skip connections and ordinary differential equations (ODEs). This perspective views ResNets as discretizations of continuous dynamic systems, where each layer represents a step in a numerical ODE solver. The key implication is that skip connections induce smoothness in representations between consecutive layers, constraining how drastically the network can transform its inputs at each step. Further work has explored the spectral properties of networks with skip connections. Fourier-based analyses [Rahaman et al., 2019, Tancik et al., 2020] have shown that these networks tend to prioritize low-frequency components, which correspond to smoother transformations. This frequency-domain perspective provides additional evidence for the smoothness-inducing nature of skip connections. Several studies have also investigated the optimization advantages provided by skip connections. The gradient stability properties of these architectures [Miyato et al., 2018] explain their trainability at extreme depths, where traditional networks struggle with vanishing or exploding gradients. However, despite these theoretical advances, the implications of skip connection-induced smoothness for parameter-efficient adaptation remain unexplored. Our work bridges this gap by connecting the smoothness properties of representations to the structure of weight adaptations, enabling more efficient parameter sharing across layers.

3 Theoretical Foundation for `CrossSpectra`

We develop a comprehensive theoretical framework explaining why neural network adaptations exhibit exploitable spectral structure across layers. Our analysis focuses on attention mechanisms in transformer architectures, revealing how skip connections induce smooth gradient fields that manifest as low-frequency patterns in weight updates. This spectral structure enables dramatic parameter reduction through frequency-domain methods. Our theoretical analysis yields three fundamental insights that directly inform `CrossSpectra`'s design:

i. **Cross-layer gradient smoothness**: Skip connections create smooth gradient propagation across transformer layers, with attention weights (Query, Key, Values) exhibiting particularly strong cross-layer correlation.

ii. **Dimension-specific spectral decay:** The spectral energy of adaptation patterns decays at different rates across dimensions; the decay is strongest along the layer axis, while spatial dimensions within each layer show a slower attenuation of high-frequency components.

iii. **Joint QKV structure**: All attention matrices ($\boldsymbol{Q}$, $\boldsymbol{K}$, $\boldsymbol{V}$) can be efficiently represented in a unified spectral framework due to their shared architectural context.

These insights motivate our unified spectral approach, where we exploit cross-layer patterns through joint parameterization of all QKV matrices.

3.1 Gradient Structure in Transformers with Skip Connections

To understand the spectral properties of attention adaptations, we first analyze how skip connections shape gradient flow through transformer architectures.

Architecture Setup. Consider a transformer with L layers. Each layer $l \in \{1, \ldots, L\}$ contains multi-head attention with weight matrices: Query $\boldsymbol{W}_l^Q \in \mathbb{R}^{d\times d}$, Key $\boldsymbol{W}_l^K \in \mathbb{R}^{d\times d}$, and Value $\boldsymbol{W}_l^V \in \mathbb{R}^{d\times d}$. The attention mechanism computes:

$$\text{Attention}_l(\boldsymbol{H}_{l-1}) = \text{softmax}\left(\frac{\boldsymbol{H}_{l-1}\boldsymbol{W}_l^Q(\boldsymbol{H}_{l-1}\boldsymbol{W}_l^K)^\top}{\sqrt{d}}\right)\boldsymbol{H}_{l-1}\boldsymbol{W}_l^V. \tag{1}$$

Crucially, skip connections define the layer-wise evolution of hidden representations:

$$\boldsymbol{H}_l = \boldsymbol{H}_{l-1} + \text{MultiHeadAttention}_l(\boldsymbol{H}_{l-1}) + \text{FFN}_l(\boldsymbol{H}_{l-1}). \tag{2}$$

This residual structure is key to understanding gradient smoothness—it ensures that gradient information flows directly across layers while being modulated by local computations.

Cross-Layer Gradient Analysis. To quantify gradient behavior across layers, we examine the gradient structure at each layer:

Definition 3.1 (Layer-wise Gradient)**.** *For a loss function $\mathcal{L}$ and attention matrices $\{\boldsymbol{W}_l^M\}_{l=1}^L$ where $M \in \{Q, K, V\}$, the gradient at layer l is:*

$$\boldsymbol{G}_l^M = \frac{\partial \mathcal{L}}{\partial \boldsymbol{W}_l^M}, \tag{3}$$

where $\boldsymbol{G}_l^M \in \mathbb{R}^{d\times d}$ has the same shape as $\boldsymbol{W}_l^M$.

Skip connections promote smooth gradient propagation across layers. Our first key result formalizes this effect under the standard residual–ODE scaling:

Theorem 3.2 (Gradient Smoothness for Attention Weights)**.** *Under residual scaling $\boldsymbol{H}_l = \boldsymbol{H}_{l-1} + \frac{1}{L}\boldsymbol{R}_l(\boldsymbol{H}_{l-1}; \boldsymbol{W}_l)$ and bounded Jacobians of $\boldsymbol{R}_l$, the gradients of attention weights satisfy*

$$\|\boldsymbol{G}_{l+1}^M - \boldsymbol{G}_l^M\|_F \le \frac{C_M}{L}, \tag{4}$$

where C_M is a depth-independent constant.

Thus, adjacent layers exhibit gradually varying gradients, and skip connections ensure this variation decays with depth, avoiding the abrupt changes typical of purely feed-forward architectures. Intuitively, the residual scaling $\frac{1}{L}$ constrains each layer's transformation to be a small perturbation of its predecessor, so gradient changes accumulate continuously along depth rather than discretely.

3.2 From Gradient Smoothness to Spectral Properties

The gradient smoothness directly translates to spectral properties of weight adaptations during fine-tuning. When gradients vary smoothly across layers, the accumulated weight changes exhibit low-frequency dominance in the layer dimension.

Adaptation Accumulation Process. During fine-tuning over T steps with learning rate η, attention weights accumulate updates:

$$\Delta \boldsymbol{W}_l^M = -\eta \sum_{t=1}^{T} \nabla_{\boldsymbol{W}_l^M} \mathcal{L}^{(t)}, \quad M \in \{Q, K, V\}. \tag{5}$$

To analyze the spectral structure, we organize all weight adaptations into a 3D tensor and examine its frequency characteristics.

Theorem 3.3 (Spectral Concentration for Attention Matrices)**.** *For attention adaptations $\{\Delta \boldsymbol{W}_l^M\}_{l=1}^L$ where $M \in \{Q, K, V\}$, define the 3D adaptation tensor $\boldsymbol{W}^M \in \mathbb{R}^{d\times d\times L}$ with $[\boldsymbol{W}^M]_{:,:,l} = \Delta \boldsymbol{W}_l^M$. Let $\hat{\boldsymbol{W}}^M(n_1, n_2, n_3)$ denote its 3D discrete Fourier transform, where $n_1 \in \{0, ..., d-1\}$ is frequency index for input dimension, $n_2 \in \{0, ..., d-1\}$ is frequency index for output dimension, and $n_3 \in \{0, ..., L-1\}$ is frequency index across layers. The Fourier coefficients exhibit dimension-specific decay with frequency:*

$$|\hat{\boldsymbol{W}}^M(n_1, n_2, n_3)| \le \frac{C_M}{(n_1+1)^{\beta_{1,M}} \cdot (n_2+1)^{\beta_{2,M}} \cdot (n_3+1)^{\beta_{3,M}}}, \tag{6}$$

where the decay rates satisfy: $\beta_{3,M} > \beta_{1,M}, \beta_{2,M}$ for all $M \in \{Q, K, V\}$.

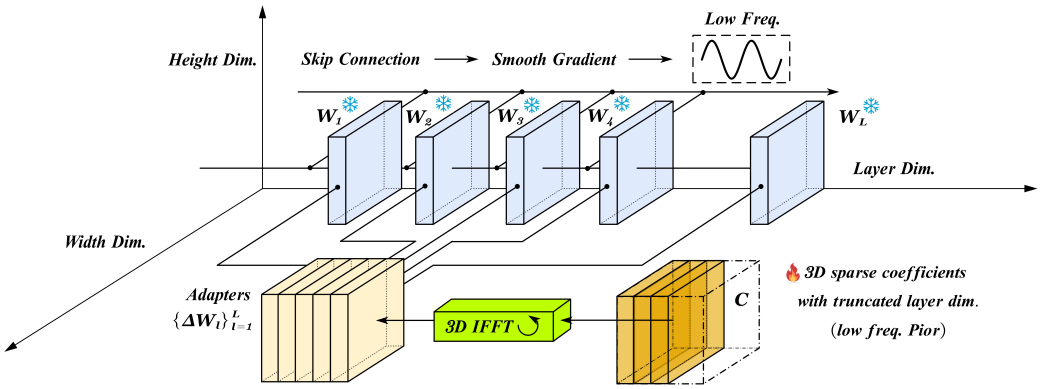


Figure 2: **The overall model**: Skip connections create smooth gradient flow across transformer layers , resulting in weight adaptations that are dominated by low-frequency patterns. `CrossSpectra` exploits this by representing all layers' adaptations as a 3D tensor and using sparse spectral coefficients via 3D inverse FFT, with truncation in the cross-layer frequency dimension.

The stronger decay in the cross-layer dimension ($\beta_{3,M} > \beta_{1,M}, \beta_{2,M}$) implies that adaptations are dominated by low-frequency patterns across layers. This is a direct consequence of the gradient smoothness induced by skip connections—smooth gradients accumulate into smooth adaptations.

4 `CrossSpectra`: From Spectral Properties to Efficient Adaptation

Building on our theoretical analysis, we now present `CrossSpectra`, a parameter-efficient fine-tuning method that directly exploits the cross-layer spectral structure in attention weight adaptations. As illustrated in Figure 2, skip connections induce smooth gradient propagation, which leads to low-frequency dominance in the learned adaptations.

From Theory to Design. Our theoretical findings directly inform three key design principles for `CrossSpectra`:

i. **Unified 3D representation**: Since gradients vary smoothly across layers, we parameterize all QKV matrices jointly rather than treating each layer independently.

ii. **Fourier decomposition with selective truncation**: The strongest spectral decay occurs in the cross-layer dimension ($\beta_3 > \beta_1, \beta_2$), so we aggressively truncate high frequencies in this dimension while preserving more frequencies within layers.

iii. **Implicit spectral regularization**: Frequency-domain parameterization naturally enforces the smoothness constraints revealed by our theoretical analysis.

These principles guide our implementation choices in the following subsections.

4.1 Unified QKV Tensor Representation

Traditional parameter-efficient methods decompose each layer's adaptations independently. For example, LoRA parameterizes:

$$\Delta \boldsymbol{W}_l^M = \boldsymbol{A}_l^M \boldsymbol{B}_l^M, \quad \boldsymbol{A}_l^M \in \mathbb{R}^{d\times r}, \boldsymbol{B}_l^M \in \mathbb{R}^{r\times d}. \tag{7}$$

This layer-wise approach cannot exploit the cross-layer patterns identified in our theoretical analysis.

`CrossSpectra` constructs a unified 3D tensor that stacks all QKV matrices across layers:

$$\boldsymbol{T}_{QKV} \in \mathbb{R}^{d\times d\times 3L}, \tag{8}$$

where the dimensions represent d for both input and output dimensions (for simplicity), and $3L$ is the extended layer dimension containing all Q, K, V matrices across L layers. This unified representation enables us to capture cross-layer patterns in a single spectral decomposition, dramatically reducing parameters compared to layer-wise methods.

4.2 Spectral Decomposition via 3D Fourier Transform

We decompose the unified QKV tensor using 3D inverse Fourier transform:

$$\boldsymbol{T}_{QKV} = \text{iFFT3D}(\boldsymbol{C}). \tag{9}$$

The coefficient tensor $\boldsymbol{C} \in \mathbb{C}^{d\times d\times 3L}$ is highly sparse, with non-zero entries denoted by the index set Ω. We control this sparsity through two key parameters: $\kappa_1 = |\Omega|/(3L)$ is number of non-zero coefficients sampled within each $d \times d$ layer's frequency space. κ_2 is number of frequencies retained in the cross-layer dimension ($\kappa_2 \ll 3L$). This parameterization yields a total parameter count of $|\Omega| = \kappa_1 \cdot \kappa_2$, resulting in a sparsity ratio of $|\Omega|/(d^2 \cdot 3L)$. For typical transformer configurations, we achieve dramatic parameter reduction by setting $|\Omega| \approx 0.1\%$ of the full tensor size. The key insight from our theory is that cross-layer spectral decay is strongest ($\beta_3 > \beta_1, \beta_2$). We exploit this by aggressive truncation in the layer dimension ($\kappa_2 \ll 3L$) while employing sparse sampling within each layer ($\kappa_1 \ll d^2$), targeting the most important frequency components.

Complex Fourier Bases with Real-Valued Output. We use complex-valued inverse 3D Fourier transforms for computational efficiency. To ensure that the reconstructed tensor $\boldsymbol{T}_{QKV} = \text{iFFT3D}(\boldsymbol{C})$ is real-valued, the sparse coefficient tensor $\boldsymbol{C}$ must satisfy discrete *Hermitian symmetry*:

$$[\boldsymbol{C}]_{u,v,w} = \overline{[\boldsymbol{C}]_{(-u) \bmod d,\ (-v) \bmod d,\ (-w) \bmod (3L)}}, \quad \forall (u,v,w) \in \Omega. \tag{10}$$

Here (u, v, w) are frequency indices along the input, output, and layer dimensions, and $\overline{(\cdot)}$ denotes complex conjugation. This condition guarantees that $\text{iFFT3D}(\boldsymbol{C})$ yields a real-number tensor. Consequently, only one half-space of frequency coefficients needs to be parameterized independently, effectively halving the number of learnable spectral parameters and reducing storage cost.

Efficient Implementation via 3D iFFT. `CrossSpectra` leverages highly optimized FFT implementations for computational efficiency. The core operations are:

i. **Forward Pass** (Algorithm 1): We first convert the sparse spectral coefficients $\boldsymbol{C}$ to the spatial domain using 3D inverse FFT, yielding the full adaptation tensor $\boldsymbol{T}_{QKV}$. This operation is performed only once per forward pass. We then extract the appropriate slice for each layer and attention matrix $(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V})$ and add it to the pre-trained weights.

ii. **Backward Pass** (Algorithm 2): We collect all gradient updates into a single tensor, transform it into the frequency domain using 3D FFT, and keep only the gradients corresponding to the sparse indices in Ω. This effectively projects the gradient into our low-dimensional spectral subspace.

This implementation ensures that computation scales with the full tensor dimensions only for the FFT operations, while parameter storage and updates remain proportional to $|\Omega|$.

Implicit Gradient Regularization. Algorithm 2 reveals that the backward pass acts as a gradient filter. When we compute $\nabla_{\boldsymbol{C}} = \text{FFT3D}(\nabla_{\boldsymbol{T}_{QKV}})$ and retain only κ_1 sparse coefficients within each layer's frequency space and κ_2 frequencies across layers, we project gradients into a subspace with low-frequency basis. This filtering prevents the accumulation of rapid variations, with the strongest regularization occurring in the cross-layer dimension where truncation is the most aggressive. The optimization process is thus biased toward discovering smooth adaptation patterns that align with the natural spectral structure revealed by our theoretical analysis.

4.3 Complexity Analysis

Table 1 provides a comprehensive comparison of memory requirements and computational costs.

Method	Memory	Forward Pass	Backward Pass
Full Fine-tuning	$\mathcal{O}(3Ld^2)$	$\mathcal{O}(Lnd^2)$	$\mathcal{O}(Lnd^2)$
LoRA (rank r)	$\mathcal{O}(6Lrd)$	$\mathcal{O}(2Lnrd)$	$\mathcal{O}(2Lnrd)$
`CrossSpectra`	$\mathcal{O}(\kappa_1\kappa_2)$	$\mathcal{O}(nd^2 + D)$	$\mathcal{O}(nd^2 + D)$

Table 1: Complexity comparison. We denote n as sequence length, L is number of layers, and $D = 3Ld^2(2\log d + \log(3L))$ is the 3D FFT cost.

Algorithm 1 `CrossSpectra` Forward Pass

Input: Sparse coefficients $\boldsymbol{C}$ with
  κ_1 non-zeros per layer, κ_2 layers.
$\boldsymbol{C}_{full} \leftarrow$ SparseToDense($\boldsymbol{C}$)
$\boldsymbol{T}_{QKV} \leftarrow$ iFFT3D($\boldsymbol{C}_{full}$)
for $l = 1$ to L **do**
  for $M \in \{Q, K, V\}$ **do**
    $\Delta \boldsymbol{W}_l^M \leftarrow$ Extract($\boldsymbol{T}_{QKV}, M, l$)
    $\tilde{\boldsymbol{W}}_l^M \leftarrow \boldsymbol{W}_l^M + \Delta \boldsymbol{W}_l^M$
  end for
end for
Return: $\{\tilde{\boldsymbol{W}}_l^M\}$

Algorithm 2 `CrossSpectra` Backward Pass

Input: Gradients $\{\nabla_{\tilde{\boldsymbol{W}}_l^M} \mathcal{L}\}$.
Initialize $\nabla_{\boldsymbol{T}_{QKV}} \in \mathbb{R}^{d \times d \times 3L}$
for $l = 1$ to L **do**
  for $M \in \{Q, K, V\}$ **do**
    Place $\nabla_{\tilde{\boldsymbol{W}}_l^M} \mathcal{L}$ into
      $\nabla_{\boldsymbol{T}_{QKV}}$ at (M, l)
  end for
end for
$\nabla_{\boldsymbol{C}_{full}} \leftarrow$ FFT3D($\nabla_{\boldsymbol{T}_{QKV}}$)
$\nabla_{\boldsymbol{C}} \leftarrow$ SparseSample($\nabla_{\boldsymbol{C}_{full}}, \kappa_1, \kappa_2$)
Return: $\nabla_{\boldsymbol{C}}$

Table 2: Performance comparison of LLaMA2 7B with different methods on eight commonsense reasoning datasets. The symbol † indicates that the results are taken from [Wang et al., 2024a, Zhong et al., 2024, Si et al., 2024].

Method	# Params(%)	BoolQ	PIQA	SIQA	HellaSwag	WinoGrande	ARC-e	ARC-c	OBQA	Average
ChatGPT †	/	73.10	85.40	68.50	78.50	66.10	89.80	79.90	74.80	77.01
LoRA†	0.84	69.80	79.90	79.50	83.60	82.60	79.80	64.70	81.00	77.61
DoRA†	0.84	71.80	83.10	79.90	89.10	83.00	84.50	71.00	81.20	80.45
PiSSA†	0.84	67.60	78.10	78.40	76.60	78.00	75.80	60.20	75.60	73.78
MiLoRA†	0.84	67.60	83.80	80.10	88.20	82.00	82.80	68.80	80.60	79.24
LoRA-Dash†	0.84	71.00	75.70	79.30	91.10	78.60	84.20	69.80	78.80	78.56
NEAT†	0.84	71.70	83.90	80.20	88.90	84.30	86.30	71.40	83.00	81.21
KaSA†	0.84	73.60	**84.40**	80.20	**91.50**	84.50	84.70	72.10	81.20	81.53
MoLoRA	0.96	73.15	83.68	80.09	74.57	85.95	87.33	72.53	86.20	80.43
HydraLoRA	0.84	72.78	84.06	79.68	80.34	86.66	87.12	72.35	86.00	81.12
`CrossSpectra`	0.02	**73.69**	83.95	**80.34**	88.42	**87.24**	**87.75**	**76.88**	**87.21**	**82.73**

The memory efficiency of `CrossSpectra` stems from sparse spectral parameterization. For typical transformer configurations with $L = 24$ layers and $d = 768$, full fine-tuning requires 42 MB, while LoRA with rank $r = 8$ needs 2.2 MB. `CrossSpectra` with $\kappa_1 = 1024$ (sparse within layers) and $\kappa_2 = 8$ (cross-layer truncation) requires only 8 KB—achieving 275× reduction compared to LoRA and 5,250× reduction compared to full fine-tuning.

Computationally, `CrossSpectra`'s forward pass consists of a single 3D iFFT operation with complexity $\mathcal{O}(3Ld^2(2\log d + \log(3L)))$, followed by standard matrix multiplications. For sequences of length n, the dominant cost remains the attention mechanism's $\mathcal{O}(n^2 d)$ complexity, making `CrossSpectra`'s FFT overhead negligible in practice. The backward pass similarly requires a 3D FFT for gradient computation with the same complexity. Modern FFT implementations on GPU further reduce this overhead through optimized memory access patterns and parallelization.

5 Experiments

Baselines. To validate the effectiveness of `CrossSpectra`, we compare against three categories of baselines: 1) **Full FT**: Full fine-tuning of all parameters; 2) **Single-LoRA variants**: LoRA [Hu et al., 2021c], DoRA [Liu et al., 2024], PiSSA [Meng et al., 2024], MiLoRA [Wang et al., 2024b], rsLoRA [Kalajdzievski, 2023], LoRA-Dash [Si et al., 2024], NEAT [Zhong et al., 2024], and KaSA [Wang et al., 2024a]; 3) **LoRA MoE methods**: MoLoRA [Zadouri et al., 2024], AdaMoLE [Liu and Luo, 2024], and HydraLoRA [Tian et al., 2024]. These baselines represent the current state of the art in parameter-efficient fine-tuning, both for standard and mixture-of-experts variants.

Benchmarks. To demonstrate the cross-modal effectiveness of our spectral approach, we evaluate `CrossSpectra` on four diverse tasks. **Image Classification (IC):** We fine-tune CLIP ViT-B/32 [Radford et al., 2021] on 7 standard image datasets including StanfordCars, DTD, EuroSAT, GTSRB, RESISC45, SUN397, and SVHN [Ilharco et al., 2023]. This evaluates `CrossSpectra`'s ability

Table 3: Performance comparison of RoBERTa-large with different methods on 7 GLUE tasks. Total rank is set to 32.

Method	# Params (%)	CoLA	SST-2	MRPC	QQP	MNLI	QNLI	RTE	Average
Full FT	100	84.27	95.98	85.29	91.58	89.83	94.49	84.84	89.47
LoRA	4.00	83.41	95.64	83.33	90.06	89.00	93.28	84.47	88.46
DoRA	4.00	85.33	95.99	84.07	91.24	89.52	93.54	84.48	89.17
PiSSA	4.00	69.12	95.98	82.84	91.24	88.94	93.59	73.29	85.00
MiLoRA	4.00	84.65	96.10	86.02	91.33	89.51	94.12	84.83	89.51
rsLoRA	4.00	83.51	95.98	86.02	90.75	88.97	93.84	84.12	89.03
MoLoRA	4.50	83.94	96.10	87.75	91.45	89.36	93.90	84.11	89.52
AdaMoLE	4.56	83.99	95.76	**86.03**	**91.48**	89.21	93.64	83.75	89.12
HydraLoRA	2.75	83.89	95.52	85.04	91.02	89.34	93.87	81.22	88.56
`CrossSpectra`	0.01	**86.86**	**96.21**	84.55	91.40	**89.55**	**94.19**	**85.56**	**89.76**

to capture visual adaptation patterns. **Natural Language Understanding (NLU):** We fine-tune RoBERTa-large [Liu, 2019] on the GLUE benchmark [Raffel et al., 2020b], which comprises diverse language tasks including grammatical acceptance (CoLA), sentiment analysis (SST-2), paraphrase detection (MRPC, QQP), and natural language inference (MNLI, QNLI, RTE). **Commonsense Reasoning (CR):** We use LLaMA2-7B [Touvron et al., 2023] on 8 reasoning benchmarks: BoolQ, PIQA, SIQA, HellaSwag, WinoGrande, ARC-e, ARC-c, and OBQA. Following Hu et al. [2023], we combine training datasets from all tasks and evaluate on each test set separately. **Arithmetic Reasoning (AR):** Using LLaMA2-7B, we evaluate mathematical reasoning capabilities on GSM8K [Cobbe et al., 2021], MAWPS [Koncel-Kedziorski et al., 2016], SVAMP [Patel et al., 2021], and AQuA [Ling et al., 2017] benchmarks. The training data combines these sources with step-by-step rationales. These diverse tasks let us verify that the cross-layer spectral structure we exploit exists across various model architectures and domains.

Implementation Details. For `CrossSpectra`, we adapt all query, key, and value projection matrices in transformer attention blocks, except in image classification tasks where we only adapt query and key matrices following standard practice. For frequency sparsity, we set the number of non-zero coefficients $|\Omega| = 3000$ (corresponding to approximately $k_1 = 1000$ samples per layer slice and $k_2 = 3$ frequencies in the layer dimension). This represents just 0.1-0.5% of the full parameter space depending on model size. For baseline comparisons, we use LoRA with rank $r = 16$ and $r = 32$. All models are trained using Adam optimizer [Kingma and Ba, 2014] with batch size 64 and cosine learning rate scheduling. For image classification, we use separate learning rates: $1e^{-3}$ for the classification layer and $1e^{-5}$ for adaptation parameters.

5.1 Main Results

Table 4: We evaluate CLIP ViT-B/32 with full fine-tuning and LoRA variants with total rank 8 across StanfordCars, DTD, EuroSAT, GTSRB, RESISC45, SUN397, and SVHN datasets. **Bold** indicates the highest results.

Method	# Params (%)	Cars	DTD	EuroSAT	GTSRB	RESISC45	SUN397	SVHN	Average
Full FT	100	60.33	73.88	98.96	98.30	93.65	53.84	96.78	82.25
LoRA	1.49	41.02	70.15	98.66	96.51	90.38	47.51	95.39	77.09
LoRA (rank16)	2.99	46.51	72.07	98.74	98.04	92.08	51.63	96.00	79.30
LoRA (rank32)	5.98	50.13	72.87	98.88	98.13	92.87	53.65	96.55	80.44
DoRA	1.49	40.75	71.91	**98.89**	97.71	90.19	47.54	95.46	77.49
PiSSA	1.49	40.41	69.62	98.48	95.84	90.58	47.21	95.84	76.85
MiLoRA	1.49	39.77	70.48	98.19	97.52	89.92	45.38	95.49	76.68
MoLoRA	2.24	50.83	73.51	98.63	97.72	92.58	52.55	96.00	80.26
AdaMoLE	2.33	49.47	71.65	98.52	97.73	91.95	52.29	95.82	79.63
HydraLoRA	1.58	48.42	72.18	98.40	97.28	92.93	51.80	96.06	79.58
`CrossSpectra`	0.03	**53.50**	**75.32**	98.82	**98.17**	**93.46**	**54.53**	**96.62**	**81.49**

The experimental results across multiple modalities in Tables 2,3,4 and 6 demonstrate that `CrossSpectra`'s superior effectiveness in parameter-efficient fine-tuning. On commonsense reasoning tasks with LLaMA2-7B, CrossSpectra achieves 82.73% average accuracy, outperforming methods

Table 5: Performance comparison under different frequency sparsity levels. `CrossSpectra` achieves optimal performance with only 3.2‰ of the frequency space.

	CSR Task				IC Task			
# Freq. $\lvert\Omega\rvert$	10000	30000	60000	120000	1000	3000	6000	12000
Sparsity $\lvert\Omega\rvert/(d^2)$	0.9‰	1.8‰	3.6‰	7.2‰	2.5‰	5.0‰	10.1‰	20.2‰
`CrossSpectra`	**82.15**	**82.73**	**82.33**	**82.65**	**79.18**	**80.34**	**81.49**	**81.32**

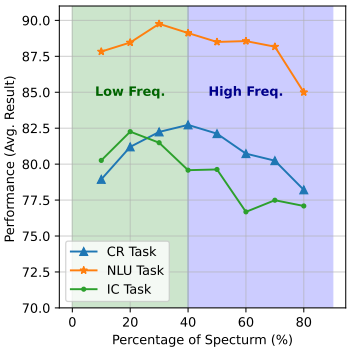


Table 6: Accuracy comparison of various LLMs using PEFT methods on arithmetic reasoning tasks. Results marked with an asterisk ($*$) are sourced from Hu et al. [Hu et al., 2023]. (†) denotes our reproduced results on LoRA.

Model	Methods	MAWPS	SVAMP	GSM8K	AQuA	Avg.
LLaMA2-7B	Base	51.7	32.4	15.7	16.9	24.8
	LoRA*	79.0	52.1	37.5	18.9	44.6
	DoRA	79.2	52.0	37.3	17.4	43.1
	PiSSA	78.6	52.5	38.0	17.7	44.7
	MiLoRA	79.0	52.1	37.5	18.9	44.6
	`CrossSpectra`	81.4	51.8	38.2	18.1	43.7

Figure 3: Left: Performance *vs.* spectrum coverage across modalities. Task performance rapidly increases with low frequencies (0-20%) and shows diminishing returns beyond 60%, validating our theoretical prediction of low-frequency dominance.

such as KaSA (81.53%) and NEAT (81.21%). For NLU tasks using RoBERTa-large, it scores 89.76% across GLUE benchmarks, excelling particularly on complex tasks such as CoLA (86.86%). The cross-modality effectiveness is further validated in vision tasks with CLIP ViT-B/32, CrossSpectra reaches 81.49% average accuracy, surpassing even high-rank LoRA variants (80.44%). For arithmetic reasoning, it outperforms standard approaches on benchmarks such as GSM8K and MAWPS. These consistent results across diverse tasks validate our theoretical framework that exploiting cross-layer smoothness via tensor-based Fourier parameterization enables substantial parameter reduction while maintaining or improving performance, confirming that weight adaptations exhibit the spectral bias predicted by our gradient analysis.

5.2 Analysis of Spectral Properties in Weight Adaptations.

The experimental analysis of Figure 1 and Figure 3 provide compelling evidence for our theoretical claims about the spectral properties of weight adaptations in neural networks with skip connections.

Low-Frequency Dominance in Adaptation Energy. Figure 1 illustrates the distribution of energy across different frequency components in weight of LLaMA2-7B (we use value projection in the attention layer for illustration). The analysis reveals that a significant portion of the total adaptation energy (69.7%) is concentrated in the low-frequency region of the spectrum, while only 30.3% is distributed across high-frequency components. This striking imbalance confirms weight adaptations should inherit the spectral properties with energy concentrated in low-frequency components.

Performance Correlation with Spectral Components. Figure 3 demonstrates how model performance across different tasks (Commonsense Reasoning, Natural Language Understanding, and Image Classification) varies as we progressively increase the sample portions of the frequency spectrum. (*i.e.*, the x-axis indicate we limit the selected frequency basis reside in the first $X\%$ part of spectrum). It demonstrates that across all three modalities (Commonsense Reasoning, Natural Language Understanding, and Image Classification), performance increases rapidly with just the initial low-frequency components ($0 - 20\%$ of the spectrum), stabilizes in the middle range ($20 - 60\%$), and shows diminishing returns in the high-frequency range (beyond 60%). The consistent pattern across diverse modalities confirms that the smoothness induced by skip connections creates adaptation patterns that are efficiently representable in the low frequency domain along the layer dimension, allowing our approach to achieve strong performance while using dramatically fewer parameters.

Table 7: **Spectral energy *vs.* model scale.** Low-frequency dominance persists across sizes.

Model	Layers	Params	Low-Freq (%)	High-Freq (%)
LLaMA2-13B	40	13B	61.2	38.8
LLaMA2-7B	32	7B	69.7	30.3
RoBERTa-L	24	355M	68.4	31.6
ViT-B/32	12	86M	70.9	29.1

Table 8: **LLaMA2-13B commonsense results.** `CrossSpectra` surpasses LoRA with far fewer parameters.

Method	BoolQ	PIQA	SIQA	HellaSwag	Avg.
LoRA	75.2	83.4	82.1	89.8	82.62
`CrossSpectra`	**76.1**	83.2	**82.7**	**92.1**	**83.52**

Table 9: **Computational efficiency.** Sub-linear time scaling with model size.

Model	Layers	Time/Epoch (s)	Params(%)	Avg.
LLaMA2-13B (`CrossSpectra`)	40	1.82	0.02	83.52
LLaMA2-7B (`CrossSpectra`)	32	0.79	0.02	82.73
LLaMA2-7B (LoRA)	32	0.61	0.84	77.61

Table 10: **FFT overhead.** 3D iFFT cost is negligible *vs.* attention.

Tensor Size	FFT Time (s)
$(4096, 4096, 3\times10)$	0.05
$(4096, 4096, 3\times20)$	0.12
$(4096, 4096, 3\times30)$	0.28
$(4096, 4096, 3\times40)$	0.41

Sparsity along the Weight Dimension. Table 6 presents a comprehensive comparison of various PEFT methods' performance specifically on arithmetic reasoning tasks across different LLM architectures (LLaMA2-7B). The evaluation spans four key mathematical benchmarks: MAWPS (math word problems), SVAMP (simple variations on arithmetic problems), GSM8K (grade school math), and AQuA (arithmetic questions and answers). The results demonstrate that `CrossSpectra` achieves comparable or superior performance to standard LoRA and DoRA methods in solving these complex mathematical problems.

Impact of Frequency Sparsity. Table 5 demonstrates how `CrossSpectra`'s performance varies with different levels of frequency sparsity across modalities. For commonsense reasoning (CSR) tasks, we tested sparsity levels from 0.9‰ to 7.2‰ of the full frequency space (10,000 to 120,000 non-zero frequencies). Similarly, for image classification (IC) tasks, we explored sparsity levels from 2.5‰ to 20.2‰ (1,000 to 12,000 frequencies). Remarkably, `CrossSpectra` achieves optimal performance with extremely sparse parameterization—only 1.8‰ (30,000 frequencies) for CSR tasks and 5.0‰ (3,000 frequencies) for IC tasks. This confirms our theoretical prediction that weight adaptations naturally concentrate in a small subset of frequency components. The consistent performance across wide sparsity ranges further validates that our approach captures the essential adaptation information while eliminating redundancy. Even at the sparsest setting (0.9‰), `CrossSpectra` outperforms traditional methods that use orders of magnitude more parameters, demonstrating the practical impact of our theoretical insights about spectral concentration in transformer adaptations.

Scaling to Larger Models. We examine whether the low-frequency concentration holds as model size increases. Table 7 shows that the low-frequency share of adaptation energy remains high (61–71%) from ViT-B/32 (86M) up to LLaMA2-13B (13B), supporting our scale-agnostic hypothesis. We further fine-tune LLaMA2-13B on commonsense reasoning. `CrossSpectra` surpasses LoRA with far fewer parameters (Table 8), indicating that our cross-layer spectral parameterization scales favorably.

Computational Efficiency. Training time scales sub-linearly with model size due to sparse spectral coefficients (Table 9). LoRA is faster per epoch but yields lower accuracy, illustrating an efficiency–accuracy trade-off. Using `torch.fft.ifftn`, 3D FFT runtime grows as $\mathcal{O}(N \log N)$ and remains negligible relative to attention (Table 10).

6 Conclusions

This work connects neural network theory with efficient adaptation methods, revealing how architectural properties determine optimal weight adaptation parameterization. `CrossSpectra` demonstrates that skip connections create structured adaptation patterns across layers—an insight previously overlooked. Future research could explore dynamic spectral bases, adaptation for emerging architectures, or information-theoretic compression limits. Our findings suggest the most efficient neural network parameterizations align with their intrinsic dynamics rather than treating parameters independently. As foundation models grow, leveraging these inherent structures will be essential for making specialized adaptation accessible.

Acknowledgments

The research is supported, in part, by the Ministry of Education, Singapore, under its Academic Research Fund Tier 1 (RG101/24); the RIE2025 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) (Award I2301E0026), administered by A*STAR, as well as supported by Alibaba Group and NTU Singapore through Alibaba-NTU Global e-Sustainability CorpLab (ANGEL). Piotr Koniusz and Hao Zhu are supported by CSIRO's Science Digital.

References

T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1

R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018. 3

K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, C. Hesse, and J. Schulman. Training verifiers to solve math word problems, 2021. 8

J. Dong, P. Koniusz, L. Feng, Y. Zhang, H. Zhu, W. Liu, X. Qu, and Y.-S. Ong. Robustifying zero-shot vision language models by subspaces alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21037–21047, 2025a. 1

J. Dong, P. Koniusz, X. Qu, and Y.-S. Ong. Stabilizing modality gap & lowering gradient norms improve zero-shot adversarial robustness of vlms. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, KDD '25, page 236–247, New York, NY, USA, 2025b. Association for Computing Machinery. ISBN 9798400712456. doi: 10.1145/3690624.3709296. 1

J. Dong, P. Koniusz, Y. Zhang, H. Zhu, W. Liu, X. Qu, and Y.-S. Ong. Improving zero-shot adversarial robustness in vision-language models by closed-form alignment of adversarial path simplices. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 14061–14078. PMLR, 13–19 Jul 2025c. 1

J. Dong, H. Zhu, Y. Zhang, X. Qu, Y.-S. Ong, and P. Koniusz. Machine unlearning via task simplex arithmetic. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025d. 1

T. Fan, H. Gu, X. Cao, C. S. Chan, Q. Chen, Y. Chen, Y. Feng, Y. Gu, J. Geng, B. Luo, et al. Ten challenging problems in federated foundation models. *IEEE Transactions on Knowledge and Data Engineering*, 2025. 1

T. Gao, A. Fisch, and D. Chen. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*, 2020. 3

Z. Gao, Q. Wang, A. Chen, Z. Liu, B. Wu, L. Chen, and J. Li. Parameter-efficient fine-tuning with discrete fourier transform. In *Proceedings of the 41st International Conference on Machine Learning*, pages 14884–14901, 2024. 3

Z. Guo, Y. Zhang, Z. Zhang, Z. Xu, and I. King. Fedlfc: Towards efficient federated multilingual modeling with lora-based language family clustering. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1519–1528, 2024. 1

J. He, C. Zhou, X. Ma, T. Berg-Kirkpatrick, and G. Neubig. Towards a unified view of parameter-efficient transfer learning. *arXiv preprint arXiv:2110.04366*, 2021. 2

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2, 3

P. He, X. Liu, J. Gao, and W. Chen. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*, 2020. 1

E. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021a. 2

E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021b. 3

E. J. Hu, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2021c. 7

Z. Hu, L. Wang, Y. Lan, W. Xu, E.-P. Lim, L. Bing, X. Xu, S. Poria, and R. K.-W. Lee. Llm-adapters: An adapter family for parameter-efficient fine-tuning of large language models. *arXiv preprint arXiv:2304.01933*, 2023. 8, 9

G. Ilharco, M. T. Ribeiro, M. Wortsman, S. Gururangan, L. Schmidt, H. Hajishirzi, and A. Farhadi. Editing models with task arithmetic, 2023. URL https://arxiv.org/abs/2212.04089. 7

D. Kalajdzievski. A rank stabilization scaling factor for fine-tuning with lora, 2023. URL https://arxiv.org/abs/2312.03732. 2, 7

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 8

R. Koncel-Kedziorski, S. Roy, A. Amini, N. Kushman, and H. Hajishirzi. Mawps: A math word problem repository. In *Proceedings of the 2016 conference of the north american chapter of the association for computational linguistics: human language technologies*, pages 1152–1157, 2016. 8

B. Lester, R. Al-Rfou, and N. Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021. 2

X. L. Li and P. Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021. 3

W. Ling, D. Yogatama, C. Dyer, and P. Blunsom. Program induction by rationale generation: Learning to solve and explain algebraic word problems. *arXiv preprint arXiv:1705.04146*, 2017. 8

H. Liu, C. Li, Y. Li, and Y. J. Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023a. 1

H. Liu, C. Li, Q. Wu, and Y. J. Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023b. 1

L. Liu, X. Liu, J. Gao, W. Chen, and J. Han. Understanding the difficulty of training transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5747–5763, 2020. 15

S.-y. Liu, C.-Y. Wang, H. Yin, P. Molchanov, Y.-C. F. Wang, K.-T. Cheng, and M.-H. Chen. Dora: Weight-decomposed low-rank adaptation. In *Forty-first International Conference on Machine Learning*, 2024. 2, 7

Y. Liu. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 8

Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 1

Z. Liu and J. Luo. AdamoLE: Fine-tuning large language models with adaptive mixture of low-rank adaptation experts. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=ndY9qFf9Sa. 7

F. Meng, Z. Wang, and M. Zhang. PiSSA: Principal singular values and singular vectors adaptation of large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=6ZBHIEtdP4. 2, 7

T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018. 3

Y. Ni, S. Zhang, and P. Koniusz. PACE: marrying the generalization of PArameter-efficient fine-tuning with consistency regularization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 2, 3

A. Patel, S. Bhattamishra, and N. Goyal. Are nlp models really able to solve simple math word problems? *arXiv preprint arXiv:2103.07191*, 2021. 8

X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 63(10):1872–1897, 2020. 1

A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 1

A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision, 2021. URL https://arxiv.org/abs/2103.00020. 1, 7

C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020a. 1

C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020b. 8

N. Rahaman, A. Baratin, D. Arpit, F. Draxler, M. Lin, F. Hamprecht, Y. Bengio, and A. Courville. On the spectral bias of neural networks. In *International conference on machine learning*, pages 5301–5310. PMLR, 2019. 2, 3, 15, 16

S.-A. Rebuffi, H. Bilen, and A. Vedaldi. Learning multiple visual domains with residual adapters. *Advances in neural information processing systems*, 30, 2017. 2

C. Si, Z. Shi, S. Zhang, X. Yang, H. Pfister, and W. Shen. Unleashing the power of task-specific directions in parameter efficient fine-tuning, 2024. URL https://arxiv.org/abs/2409.01035. 2, 7

A. Singh, R. Hu, V. Goswami, G. Couairon, W. Galuba, M. Rohrbach, and D. Kiela. Flava: A foundational language and vision alignment model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15638–15650, 2022. 1

M. Tancik, P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. Barron, and R. Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33:7537–7547, 2020. 3

C. Tian, Z. Shi, Z. Guo, L. Li, and C. Xu. Hydralora: An asymmetric lora architecture for efficient fine-tuning, 2024. URL https://arxiv.org/abs/2404.19245. 7

H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 8

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2, 3

F. Wang, J. Jiang, C. Park, S. Kim, and J. Tang. Kasa: Knowledge-aware singular-value adaptation of large language models, 2024a. URL https://arxiv.org/abs/2412.06071. 2, 7

H. Wang, Y. Li, S. Wang, G. Chen, and Y. Chen. Milora: Harnessing minor singular components for parameter-efficient llm finetuning, 2024b. URL https://arxiv.org/abs/2406.09044. 2, 7

R. Xiong, Y. Yang, D. He, K. Zheng, S. Zheng, C. Xing, H. Zhang, Y. Lan, L. Wang, and T. Liu. On layer normalization in the transformer architecture. In *International conference on machine learning*, pages 10524–10533. PMLR, 2020. 15

M. Yang, J. Chen, Y. Zhang, J. Liu, J. Zhang, Q. Ma, H. Verma, Q. Zhang, M. Zhou, I. King, et al. Low-rank adaptation for foundation models: A comprehensive review. *arXiv preprint arXiv:2501.00365*, 2024. 1

T. Zadouri, A. Üstün, A. Ahmadian, B. Ermis, A. Locatelli, and S. Hooker. Pushing mixture of experts to the limit: Extremely parameter efficient moe for instruction tuning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=EvDeiLv7qc. 7

Q. Zhang, M. Chen, A. Bukharin, P. He, Y. Cheng, W. Chen, and T. Zhao. Adaptive budget allocation for parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.10512*, 2023. 3

S. Zhang, A. Chen, Y. Sun, J. Gu, Y.-Y. Zheng, P. Koniusz, K. Zou, A. van den Hengel, and Y. Xue. Primitive vision: Improving diagram understanding in MLLMs. In *Forty-second International Conference on Machine Learning*, 2025a. 1

S. Zhang, Y. Ni, J. Du, Y. Xue, P. Torr, P. Koniusz, and A. van den Hengel. Open-world objectness modeling unifies novel object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 30332–30342, June 2025b. 1

Y. Zhang, D. Zeng, J. Luo, X. Fu, G. Chen, Z. Xu, and I. King. A survey of trustworthy federated learning: Issues, solutions, and challenges. *ACM Transactions on Intelligent Systems and Technology*, 15(6):1–47, 2024a. 1

Y. Zhang, H. Zhu, A. Liu, H. Yu, P. Koniusz, and I. King. Less is more: Extreme gradient boost rank-1 adaption for efficient finetuning of llms. *arXiv preprint arXiv:2410.19694*, 2024b. 3

Y. Zhang, H. Zhu, Z. Song, Y. Chen, X. Fu, Z. Meng, P. Koniusz, and I. King. Geometric view of soft decorrelation in self-supervised learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4338–4349, 2024c. 1

Y. Zhang, H. Zhu, A. Z. Tan, D. Yu, L. Huang, and H. Yu. pfedmxf: Personalized federated class-incremental learning with mixture of frequency aggregation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 30640–30650, 2025c. 3

Y. Zhong, H. Jiang, L. Li, R. Nakada, T. Liu, L. Zhang, H. Yao, and H. Wang. Neat: Nonlinear parameter-efficient adaptation of pre-trained models, 2024. URL https://arxiv.org/abs/2410.01870. 2, 7

H. Zhu, Y. Zhang, J. Dong, and P. Koniusz. BiLoRA: almost-orthogonal parameter spaces for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 25613–25622, June 2025. 3

A Proofs

Our analysis builds on standard stability and smoothness properties of residual architectures.

Assumption A.1 (Residual stability and ODE scaling)**.** *Each residual block $\boldsymbol{R}_l(\boldsymbol{H}_{l-1}; \boldsymbol{W}_l)$ is Lipschitz continuous with $\|\partial \boldsymbol{R}_l / \partial \boldsymbol{H}_{l-1}\|_2 \le c_H$ and $\|\partial \boldsymbol{R}_l / \partial \boldsymbol{W}_l^M\|_2 \le c_M$, where c_H, c_M are depth-independent constants. We assume residual–ODE scaling $\boldsymbol{H}_l = \boldsymbol{H}_{l-1} + \frac{1}{L}\boldsymbol{R}_l(\boldsymbol{H}_{l-1}; \boldsymbol{W}_l)$, and bounded loss gradients $\|\partial \mathcal{L} / \partial \boldsymbol{H}_L\|_F \le G$ (Xiong et al., 2020, Liu et al., 2020). Layer normalization and residual scaling guarantee such bounded Jacobians.*

Assumption A.2 (Spatial regularity and weak separability)**.** *Within-layer adaptation maps possess bounded Sobolev norms, implying decay of high-frequency spectral components in each spatial dimension. Such spectral bias is well documented in neural networks (Rahaman et al., 2019). Moreover, the 3-D adaptation tensor is approximately separable across spatial and layer dimensions, as suggested by empirical analyses of attention weight spectra.*

These assumptions are mild and supported by empirical evidence in prior work. They ensure that the subsequent proofs yield depth-normalized gradient smoothness and power-law spectral decay.

A.1 Proof of Theorem 3.2

Proof. **Step 1: Residual formulation.** We express each transformer block in residual–ODE form $\boldsymbol{H}_l = \boldsymbol{H}_{l-1} + \frac{1}{L}\boldsymbol{R}_l(\boldsymbol{H}_{l-1}; \boldsymbol{W}_l)$, where $\boldsymbol{R}_l$ denotes the combined attention and feed-forward submodules. Let $J_j = \partial \boldsymbol{R}_j / \partial \boldsymbol{H}_{j-1}$ and $K_l = \partial \boldsymbol{R}_l / \partial \boldsymbol{W}_l^M$, with $\|J_j\|_2 \le c_H$ and $\|K_l\|_2 \le c_M$.

Step 2: Gradient decomposition. By the chain rule,

$$\nabla_{\boldsymbol{W}_l^M} \mathcal{L} = \frac{\partial \mathcal{L}}{\partial \boldsymbol{H}_L} \frac{\partial \boldsymbol{H}_L}{\partial \boldsymbol{W}_l^M}, \qquad \frac{\partial \boldsymbol{H}_L}{\partial \boldsymbol{W}_l^M} = \frac{1}{L} \sum_{t=l}^{L} \Phi_{t+1 \leftarrow l} K_l,$$

where the propagation operator $\Phi_{t+1 \leftarrow l} = \prod_{j=l+1}^{t} (I + \frac{1}{L} J_j)$ satisfies $\|\Phi_{t+1 \leftarrow l}\|_2 \le e^{c_H (t-l)/L}$.

Step 3: Bounding cross-layer differences. Successive gradients differ by

$$\frac{\partial \boldsymbol{H}_L}{\partial \boldsymbol{W}_{l+1}^M} - \frac{\partial \boldsymbol{H}_L}{\partial \boldsymbol{W}_l^M} = \frac{1}{L} \sum_{t=l+1}^{L} (\Phi_{t+1 \leftarrow l+1} - \Phi_{t+1 \leftarrow l}) K_l.$$

Using the series expansion $\Phi_{t+1 \leftarrow l+1} - \Phi_{t+1 \leftarrow l} = O(\frac{1}{L})$ and $\|\Phi_{t+1 \leftarrow l}\|_2 \le e^{c_H (t-l)/L}$ gives

$$\left\| \frac{\partial \boldsymbol{H}_L}{\partial \boldsymbol{W}_{l+1}^M} - \frac{\partial \boldsymbol{H}_L}{\partial \boldsymbol{W}_l^M} \right\|_2 \le \frac{(e^{c_H} - 1) c_M}{L}.$$

Multiplying by $\|\partial \mathcal{L} / \partial \boldsymbol{H}_L\|_F \le G$ yields

$$\|\boldsymbol{G}_{l+1}^M - \boldsymbol{G}_l^M\|_F \le \frac{C_M}{L}, \qquad C_M = G\, c_M (e^{c_H} - 1).$$

□

A.2 Proof of Theorem 3.3

Proof. **Step 1: Constructing the adaptation tensor.** Let $\Delta \boldsymbol{W}_l^M$ denote the accumulated weight updates, and form $\boldsymbol{A}^M \in \mathbb{R}^{d \times d \times L}$ with $\boldsymbol{A}_{:,:,l}^M = \Delta \boldsymbol{W}_l^M$. Define its 3D discrete Fourier transform:

$$\widehat{\boldsymbol{A}}^M(n_1, n_2, n_3) = \sum_{i_1, i_2, l} \boldsymbol{A}_{i_1, i_2, l}^M \, e^{-2\pi i (n_1 i_1 / d + n_2 i_2 / d + n_3 l / L)}.$$

Step 2: Smoothness implies spectral decay. From Theorem 3.2,

$$\|\Delta \boldsymbol{W}_{l+1}^M - \Delta \boldsymbol{W}_l^M\|_F \le \frac{C_1}{L}.$$

A bounded first finite difference yields Fourier magnitude

$$\|\widehat{\boldsymbol{A}}^M_{:,:,n_3}\|_F \leq \frac{\tilde{C}_1}{1+n_3}.$$

If a stronger second-difference bound $\|\Delta \boldsymbol{W}^M_{l+1} - 2\Delta \boldsymbol{W}^M_l + \Delta \boldsymbol{W}^M_{l-1}\|_F \leq C_2/L^2$ holds, then

$$\|\widehat{\boldsymbol{A}}^M_{:,:,n_3}\|_F \leq \frac{\tilde{C}_2}{(1+n_3)^2}.$$

This confirms a power-law decay in the cross-layer dimension.

Step 3: Combined spatial–layer spectral structure. Empirically, neural networks exhibit spectral bias toward low within-layer frequencies [Rahaman et al., 2019], so that

$$|\widehat{\boldsymbol{A}}^M(n_1, n_2, n_3)| \leq \frac{C_M}{(1+n_1)^{\beta_1}\,(1+n_2)^{\beta_2}\,(1+n_3)^{\beta_3}},$$

where $\beta_3 > \beta_1, \beta_2$ due to stronger smoothness along depth. This establishes the claimed dimension-specific spectral decay.

□

B Limitations and Future Work

While `CrossSpectra` demonstrates significant parameter efficiency across diverse tasks, there are several limitations worth noting. First, our current approach focuses exclusively on adapting attention weights $(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V})$, potentially missing optimization opportunities in other components like feed-forward networks. This design choice has the benefit of reducing overfitting risk by targeting the most information-dense parameters identified by our gradient analysis, but extending the spectral approach to other transformer components in a principled way could yield further improvements. Second, the computational overhead of 3D FFT operations, while well-optimized on modern hardware, might become a bottleneck for extremely large models (trillions of parameters) or resource-constrained deployment environments. Third, our method requires a careful selection of frequency sparsity patterns—currently uniform sampling within layers—which may not be optimal for all model architectures or tasks. A promising direction for future work is to develop adaptive frequency sampling strategies that automatically identify the most important spectral components for a specific task. Additionally, exploring theoretical connections between `CrossSpectra` and other parameter-efficient methods like prompt tuning could lead to hybrid approaches that combine their complementary strengths. Finally, while we observed consistent cross-layer spectral patterns across model sizes up to 7B parameters, verifying that these properties scale efficiently to models with hundreds of billions of parameters remains an important direction for future research.

NeurIPS Paper Checklist

i. **Claims**

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

Answer: [Yes] ,

Justification: We introduce contributions and scope on introduction.

Guidelines:

- The answer NA 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 NA 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.

ii. **Limitations**

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

Answer: [Yes]

Justification: We discuss the limitation of our work on the Appendix

Guidelines:

- The answer NA 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.

iii. **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:We provide the full set of assumptions and a roof

Guidelines:

- The answer NA 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.

iv. **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: We write all the main experimental results

Guidelines:

- The answer NA 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.

v. **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: [No]

Justification: We promise code will be public available when paper get accepted

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.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://nips.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.

vi. **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, etc.) necessary to understand the results?

Answer: [Yes]

Justification: The results and experiment are displayed in experiment part

Guidelines:

- The answer NA 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.

vii. **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: We define the standard deviation over five times training, thereby conveying statistical significance and variability.

Guidelines:

- The answer NA 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.

viii. **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 details of this experiment are displayed in main paper

Guidelines:

- The answer NA 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).

ix. **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: This paper conducted in the paper conform with the NeurIPS Code of Ethics

Guidelines:

- The answer NA 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).

x. **Broader impacts**

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

Answer: [Yes]

Justification: paper discusses both potential positive societal impacts and negative societal impacts of the work performed

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA 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).

xi. **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., pretrained language models, image generators, or scraped datasets)?

Answer: [NA] ,

Justification: This work does not have any danger.

Guidelines:

- The answer NA 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.

xii. **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: We use all public datasets without private data.

Guidelines:

- The answer NA 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.

xiii. **New assets**

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

Answer: [NA] .

Justification: We use all public datasets without private data.

Guidelines:

- The answer NA 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.

xiv. **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: [NA] .

Justification: We do not have crowdsourcing experiments and research with human subjects.

Guidelines:

- The answer NA 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.

xv. **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: [NA] .

Justification: We do not have crowdsourcing experiments and research with human subjects.

Guidelines:

- The answer NA 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.

xvi. **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 rigorousness, or originality of the research, declaration is not required.

Answer: [NA] .

Justification: We use LLM for grammar correction.

Guidelines:

- The answer NA 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 (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.