RETHINKING FINE-TUNING: UNLOCKING HIDDEN CAPABILITIES IN VISION-LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Fine-tuning has become the dominant paradigm for adapting Vision-Language Models (VLMs), yet most approaches rely on explicit weight updates that introduce a fundamental trade-off. Full Fine-Tuning (FFT) may perturb pretrained representations due to cross-modal gradient interference, whereas Parameter-Efficient Fine-Tuning (PEFT) methods rely on additive modules, such as low-rank adapters, which may limit adaptation capacity. In this paper, we rethink VLM adaptation from a structural selection framework that adapts VLMs without modifying backbone weights, and we propose **Mask Fine-Tuning (MFT)**. MFT learns masks that selectively route information through existing pretrained connections, dynamically uncovering subnetworks that better align pretrained representations with downstream objectives. Extensive experiments show that MFT provides an effective structural alternative to both FFT and PEFT, consistently achieving superior performance across multiple vision-language benchmarks without adding knowledge or altering the deployment architecture. Moreover, our analysis with MFT provides new insights into how pretrained VLMs reorganize their internal representational pathways during adaptation.

1 INTRODUCTION

The advent of pre-trained Vision-Language Models (VLMs) (Liu et al., 2023b) represents a significant milestone in artificial intelligence (Bordes et al., 2024). By training on large-scale multimodal data, these models have demonstrated remarkable capabilities and set state-of-the-art standards across a spectrum of tasks, including visual question answering (VQA) (Agrawal et al., 2016), image captioning (Li et al., 2022), and complex multimodal learning (Li et al., 2025b). The VLM ecosystem is expanding at an unprecedented rate, with the proliferation of powerful open-weight language models (Bai et al., 2023; Touvron et al., 2023; Team et al., 2024; Gunasekar et al., 2023; Zheng et al., 2023b) built with fantastic vision backbones (Radford et al., 2021; Zhai et al., 2023). However, despite their impressive zero-shot generalization, achieving peak performance in downstream applications, such as medical diagnostics (Pham & Ngo, 2025), robotics (Xu et al., 2025), or domain-specific document understanding (Kim et al., 2022), usually requires additional adaptation.

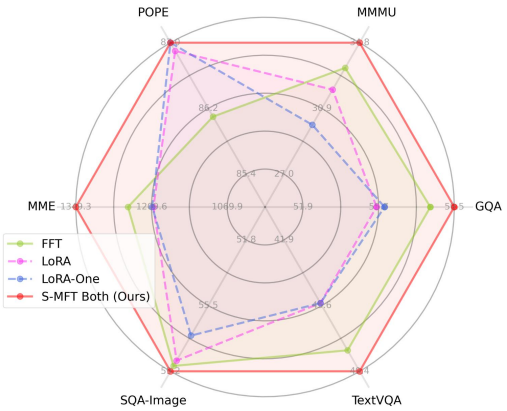

Figure 1: Comparison of fine-tuning methods on a VLM (Qwen2.5-0.5B language model).

Traditionally, Full Fine-Tuning (FFT) has been the default paradigm (Devlin et al., 2019; Ouyang et al., 2022). However, at model scale, FFT has become computationally and logistically prohibitive, requiring massive computational resources and storage for each downstream task (Zhang et al., 2025a). To address this, Parameter-Efficient Fine-Tuning (PEFT) (Zhang et al., 2025a) methods, notably Low-Rank Adaptation (LoRA) (Hu et al., 2021), have been proposed. Although these addi-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

tive PEFT paradigms balance effectiveness and efficiency, they introduce adaptation via additional low-rank modules, which may limit the capacity to model complex multimodal interactions.

In this paper, we rethink VLM adaptation not as an additive optimization problem, but as a structural selection problem over pretrained connections. While mask-based subnetwork discovery has been explored for narrow vision or language tasks (Zhao et al., 2020; Zheng et al., 2023a; Fang et al., 2024b), its potential for adapting models from pre-trained checkpoints remains largely unexplored. We introduce Mask Fine-Tuning (MFT), which selectively reorganizes the overlapping representational pathways already encoded within the frozen VLM backbone. Rather than introducing external modules, MFT adapts the model by learning static masks that attenuate selected pre-trained connections. We conduct extensive experiments across diverse VLMs. The results show that MFT consistently and significantly surpasses strong baselines. Our contributions are as follows:

- We reconceptualize masking as a structural adaptation mechanism for VLMs, framing fine-tuning as a structural selection problem over pretrained connections.
- We propose Mask Fine-Tuning (MFT), a structural selection framework that adapts VLMs through trainable mask scores while keeping the pre-trained backbone weights frozen and adding no deployment-time parameters.
- Masks learned by MFT reveal structured sparsity patterns within VLMs, providing new insights into how pretrained models reorganize internal pathways during adaptation.

2 Related Work

We contextualize Mask Fine-Tuning (MFT) within the broader landscape: 1) the evolution of fine-tuning methods for VLMs, 2) distinguishing MFT from model pruning, and 3) relevant mask-learning for fine-tuning methods.

2.1 Vision-Language Models Fine-tuning

Full Fine-Tuning (FFT) has been proven to be an effective downstream task adaptation mechanism of VLMs (Oh et al., 2024; Zhai et al., 2024). However, the prohibitive costs of FFT have become a problem as model size scales up (Zhang et al., 2025a), leading to the rise of PEFT (Zhang et al., 2025a). The vast majority of PEFT techniques, while diverse, share the assumption that adaptation requires introducing a weight update to the model. Adapter-Tuning (Houlsby et al., 2019) inserts small, bottleneck-shaped, and learnable neural modules (adapters) between the network layers. Prefix-Tuning (Li & Liang, 2021) prepends a sequence of continuous and learnable vectors to the keys and values of the attention layers. Prompt-Tuning (Lester et al., 2021) simplifies Prefix-Tuning by adding learnable embeddings only to the input layer. P-Tuning (Liu et al., 2022b) introduces a prompt encoder to more stably optimize continuous prompts. Low-Rank Adaptation (LoRA) (Hu et al., 2021) is based on the hypothesis that the update matrix has a low intrinsic rank. It freezes the pre-trained weights and injects a trainable low-rank branch. This additive weight-update philosophy is shared by variants such as DoRA (Liu et al., 2024), which decomposes the update into magnitude and direction components, and VeRA (Kopiczko et al., 2024), which shares low-rank matrices across layers. Instead of adding new parameters or updating weights to achieve adaptation, MFT adapts the model by learning masks and keeps the pre-trained weights frozen.

2.2 Pruning and Latent Subnetworks

MFT shares conceptual connections with model pruning and latent subnetworks, which examine how over-parameterized models encode adaptable functional pathways. Neural network pruning (Liu et al., 2019; Molchanov et al., 2017; Wang et al., 2022) has been widely adopted across vision (Fang et al., 2024a), language (Liu et al., 2025b), and multimodal learning (Liu et al., 2025c) for compression (Han et al., 2016b; Iandola et al., 2016) and acceleration (Han et al., 2016a; Wang et al., 2021; Wang & Fu, 2023; Ma et al., 2023; Fang et al., 2023). In parallel, the Lottery Ticket Hypothesis (LTH) (Frankle & Carbin, 2019) revealed that dense networks can contain sparse subnetworks capable of matching dense performance. The Dual Lottery Ticket Hypothesis (DLTH (Bai et al., 2022)) further showed that targeted regularization can transform even randomly selected subnetworks into trainable ones. Similar findings in pre-trained models such as BERT (Liu et al.,

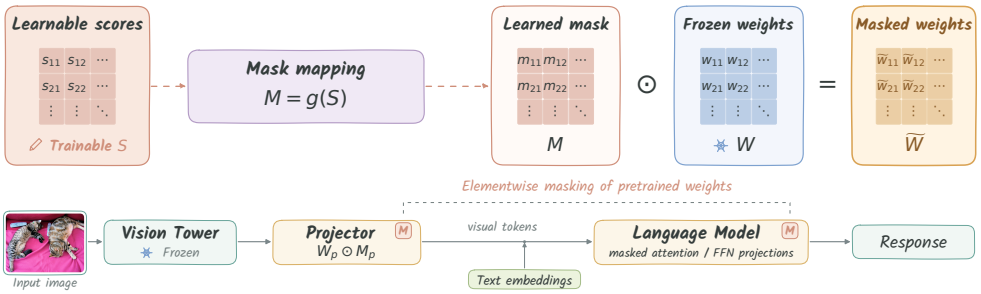

Figure 2: Overview of **Mask Fine-Tuning (MFT)**. All pretrained parameters remain frozen, while learnable scores S are optimized for the projector and selected linear layers of the language backbone. The scores are mapped to masks $M = g(S)$, which modulate the pretrained weights W through elementwise multiplication, yielding the effective weights $\widetilde{W} = W \odot M$.

2022c) indicate that transferable subnetworks exist and can be selectively activated for new tasks in large models, consistent with MFT's view of reusing existing structures in foundation models with frozen weights. While both pruning and MFT involve identifying meaningful connections, their fundamental objectives and underlying mechanisms diverge sharply. Pruning and traditional LTH searches focus strictly on model compression, aiming to reduce model size and inference latency via permanent sparsity (Frantar & Alistarh, 2023; Sun et al., 2024), often accompanied by performance degradation. In contrast, MFT treats masking not as a compression-oriented tool, but as an *end-to-end structural selection mechanism* designed to facilitate cross-modal interactions. Rather than reducing parameters, MFT is designed for knowledge elicitation, removing redundancy and improving performance.

2.3 Subnetwork and Mask Learning Adaptation

MFT belongs to a new and emerging class of methods focusing on subnetwork identification, in which adaptation is achieved solely by learning a mask over a completely frozen backbone. This paradigm reframes model adaptation as a structural selection problem rather than parameter optimization, aligning with a broader trend of identifying optimal subnetworks within pre-trained models (Fang et al., 2024b; Hou et al., 2025; Liu et al., 2025a; Tao et al., 2023). Early seminal works, such as (Zhao et al., 2020), show that discrete binary masking of pre-trained BERT weights can achieve performance comparable to FFT. Regularized Mask Tuning (RMT) (Zheng et al., 2023a) extends this idea to the vision encoder in CLIP (Radford et al., 2021). While effective at identifying isolated subnetworks for narrow, discriminative tasks, these methods are architecturally misaligned with the complex, cross-modal generative instruction-following required by modern VLMs. Furthermore, recent methods (Zhang et al., 2025b) treat masking as a strictly post-hoc enhancement step. MFT distinguishes itself as an end-to-end, general-purpose instruction-tuning strategy that operates directly on base pretrained VLMs.

3 Mask Fine-Tuning (MFT)

We adopt an MFT approach for VLMs, specifically learning masks for the language backbone and projector components, shown in Fig. 2.

3.1 Mask Parameterization

For each pre-trained weight matrix $W \in \mathbb{R}^{m \times n}$ in the language model and projector, we introduce a learnable score matrix of the same size $S \in \mathbb{R}^{m \times n}$ to help generate the mask $M \in \mathbb{R}^{m \times n}$. The effective weight is computed as:

$$W' = W \odot M, \tag{1}$$

where $\odot$ denotes element-wise multiplication, W is frozen during training.

3.2 LEARNABLE MASK

Binary (Hard) Mask. For a target sparsity level $k \in [0,1]$, set $s = \lfloor kmn \rfloor$. Let $\mathcal{I}_s(S) \subseteq \{1,\dots,m\} \times \{1,\dots,n\}$ be the set of exactly s indices having the smallest values of $|S_{ij}|$, using a fixed lexicographic rule to break ties. We define

$$M_{ij} = \begin{cases} 0, & (i,j) \in \mathcal{I}_s(S), \\ 1, & (i,j) \notin \mathcal{I}_s(S). \end{cases} \tag{2}$$

This definition masks exactly s entries and retains exactly $mn - s$. The trainable score magnitude provides a task-aware ranking signal that directly determines the selected connections.

To optimize the exact top-$(1-k)$ binary selection, we employ the Straight-Through Estimator (STE) (Bengio et al., 2013) for gradient $\mathcal{L}$ computation, which enables gradients to flow directly through the discrete masking operation by approximating the Jacobian as an identity mapping:

$$\frac{\partial M}{\partial S} \approx I, \tag{3}$$

where I denotes the identity. While the forward pass enforces strict binary values via top-k selection, the backward pass treats the operation as identity, enabling the scores S to receive meaningful gradient signals. By the chain rule:

$$\frac{\partial \mathcal{L}}{\partial S} = \frac{\partial \mathcal{L}}{\partial M} \cdot \frac{\partial M}{\partial S} \approx \frac{\partial \mathcal{L}}{\partial M} \cdot I = \frac{\partial \mathcal{L}}{\partial M}. \tag{4}$$

This approximation provides effective gradients for optimization, guiding which elements to mask.

Continuous (Soft) Mask. Binary masking provides exact sparsity and clear interpretability, while S-MFT complements it with a differentiable formulation that automatically calibrates masks using visual and linguistic signals. Based on a temperature-scaled sigmoid function, S-MFT smoothly learns task-adaptive structural selection patterns:

$$M_{ij} = \sigma\left(\frac{S_{ij}}{T}\right) = \frac{1}{1 + \exp(-S_{ij}/T)}, \tag{5}$$

where $T > 0$ is a static temperature parameter that controls the sharpness of the function. The sigmoid parameterization places every effective weight on the line segment between zero and its pre-trained value, preserving the pre-trained sign and preventing uncontrolled amplification. This structure-selection geometry supplies a focused inductive bias while retaining continuous, task-adaptive control over every selected connection.

3.3 TRAINING OBJECTIVE

MFT uses the same token-level negative log-likelihood as FFT, with the effective weights replaced by $W \odot M$:

$$\mathcal{L}(U_m; M) = -\tfrac{1}{L} \textstyle\sum_{i=1}^{L} \log P\big(u_m^i \mid u_m^{<i}; W \odot M\big). \tag{6}$$

where $U_m = \{u_m^1, \dots, u_m^L\}$ is the target token sequence. The pre-trained weights W are fixed, and we optimize the mask scores.

3.4 THEORETICAL GROUNDING

Condition on pre-trained weights W_0 fixed before an i.i.d. downstream sample $\mathcal{S}$ of size n, and let R and $\widehat{R}_{\mathcal{S}}$ denote population and empirical risk for a bounded loss $\ell \in [0,1]$. On the d mask-eligible weights, compare MFT and FFT at the same effective update radius r, and write $\mathbb{B}_2(r) = \{\Delta W : \|\Delta W\|_2 \le r\}$:

$$\begin{aligned} \mathcal{U}_{\mathrm{MFT}}(r) &= \{W_0 \odot (M - \mathbf{1}) : M \in [0,1]^d\} \cap \mathbb{B}_2(r) \\ &\subseteq \mathcal{U}_{\mathrm{FFT}}(r) = \mathbb{B}_2(r), \\ \mathcal{H}_{\mathrm{MFT}}(r) &\subseteq \mathcal{H}_{\mathrm{FFT}}(r) \end{aligned} \tag{7}$$

where $\mathcal{H}_A(r) = \{x \mapsto f(x; W_0 + \Delta W) : \Delta W \in \mathcal{U}_A(r)\}$. The inclusion follows because every MFT update obeys $\Delta W_i W_{0,i} \le 0$ and $|\Delta W_i| \le |W_{0,i}|$. Monotonicity of empirical Rademacher complexity then gives

$$\widehat{\mathfrak{R}}_n(\ell \circ \mathcal{H}_{\mathrm{MFT}}(r)) \le \widehat{\mathfrak{R}}_n(\ell \circ \mathcal{H}_{\mathrm{FFT}}(r)). \tag{8}$$

Table 1: Main experimental result comparison with the time cost of our Soft Mask Fine-Tuning (S-MFT) with Full Fine-Tuning (FFT) and representative Parameter-Efficient Fine-Tuning (PEFT) baselines across four language backbones. We repeat each experiment five times with different random seeds and report the mean and standard deviation. Red numbers denote the best results, and blue numbers denote the second-best. All results reported here are the performance after a full training epoch. "Epochs" and "Time" are reported as the efficiency metric, indicating when a method converges and achieves stable, optimal validation performance.

	Method	Evaluation Tasks						Epoch ↓	Time (h) ↓
		GQA ↑	MMMU ↑	POPE ↑	MME ↑	SQA-Image ↑	TextVQA ↑		
Qwen2.5-0.5B	Zero-Shot	5.7	22.6	49.9	93.7	3.2	9.0	-	-
	FFT	$58.3_{\pm 0.28}$	$33.3_{\pm 0.37}$	$86.1_{\pm 0.46}$	$1253.0_{\pm 10.2}$	$58.9_{\pm 0.25}$	$48.2_{\pm 0.23}$	0.90	4.91
	LoRA	$55.6_{\pm 0.28}$	$32.0_{\pm 0.35}$	$86.9_{\pm 0.45}$	$1206.0_{\pm 10.1}$	$58.6_{\pm 0.22}$	$45.5_{\pm 0.24}$	0.65	3.14
	DoRA	$57.1_{\pm 0.24}$	$32.8_{\pm 0.31}$	$87.1_{\pm 0.42}$	$1228.5_{\pm 9.4}$	$58.8_{\pm 0.21}$	$46.8_{\pm 0.28}$	0.63	3.42
	QLoRA	$53.8_{\pm 0.36}$	$30.1_{\pm 0.41}$	$85.2_{\pm 0.47}$	$1178.4_{\pm 10.2}$	$56.9_{\pm 0.27}$	$46.7_{\pm 0.39}$	0.88	5.97
	Parallel Adapter	$56.0_{\pm 0.25}$	$30.8_{\pm 0.32}$	$86.5_{\pm 0.49}$	$1215.4_{\pm 9.5}$	$57.6_{\pm 0.20}$	$45.8_{\pm 0.23}$	0.72	3.95
	P-Tuning v2	$52.8_{\pm 0.21}$	$28.5_{\pm 0.34}$	$84.1_{\pm 0.46}$	$1162.3_{\pm 10.9}$	$54.2_{\pm 0.29}$	$42.4_{\pm 0.39}$	0.85	3.58
	S-MFT Attn(Ours)	$58.0_{\pm 0.27}$	$33.3_{\pm 0.31}$	$86.7_{\pm 0.47}$	$1243.0_{\pm 8.2}$	$59.7_{\pm 0.20}$	$47.9_{\pm 0.37}$	**0.61**	**2.92**
	S-MFT MLP (Ours)	$59.0_{\pm 0.21}$	$33.4_{\pm 0.32}$	$87.3_{\pm 0.45}$	$1306.0_{\pm 8.4}$	$60.0_{\pm 0.17}$	$48.9_{\pm 0.39}$	**0.62**	**3.13**
	S-MFT Both(Ours)	$59.5_{\pm 0.22}$	$34.8_{\pm 0.34}$	$87.0_{\pm 0.48}$	$1349.3_{\pm 8.6}$	$59.2_{\pm 0.19}$	$49.4_{\pm 0.38}$	0.67	3.81
TinyLlama-1.1B	Zero-Shot	1.0	26.1	49.8	211.8	4.0	6.8	-	-
	FFT	$57.3_{\pm 0.33}$	$30.4_{\pm 0.34}$	$85.8_{\pm 0.44}$	$1180.9_{\pm 9.2}$	$60.3_{\pm 0.27}$	$47.9_{\pm 0.23}$	0.93	5.76
	LoRA	$57.5_{\pm 0.41}$	$30.4_{\pm 0.22}$	$86.9_{\pm 0.47}$	$1167.6_{\pm 9.5}$	$58.2_{\pm 0.19}$	$47.9_{\pm 0.27}$	0.77	4.53
	DoRA	$57.4_{\pm 0.31}$	$30.9_{\pm 0.25}$	$87.1_{\pm 0.43}$	$1175.2_{\pm 9.8}$	$59.8_{\pm 0.18}$	$48.1_{\pm 0.26}$	0.72	4.81
	QLoRA	$55.9_{\pm 0.35}$	$26.2_{\pm 0.35}$	$81.5_{\pm 0.48}$	$1142.8_{\pm 11.5}$	$56.5_{\pm 0.22}$	$46.2_{\pm 0.38}$	0.62	5.03
	Parallel Adapter	$56.8_{\pm 0.30}$	$29.8_{\pm 0.31}$	$86.4_{\pm 0.48}$	$1165.5_{\pm 8.9}$	$57.5_{\pm 0.21}$	$46.8_{\pm 0.24}$	0.81	4.62
	P-Tuning v2	$53.4_{\pm 0.27}$	$25.1_{\pm 0.28}$	$82.5_{\pm 0.44}$	$1105.0_{\pm 11.8}$	$52.7_{\pm 0.14}$	$44.4_{\pm 0.22}$	0.88	4.25
	S-MFT Attn(Ours)	$57.1_{\pm 0.20}$	$30.7_{\pm 0.29}$	$87.6_{\pm 0.47}$	$1170.9_{\pm 7.4}$	$60.4_{\pm 0.15}$	$49.1_{\pm 0.37}$	**0.61**	**2.98**
	S-MFT MLP (Ours)	$57.9_{\pm 0.30}$	$30.9_{\pm 0.20}$	$88.0_{\pm 0.48}$	$1237.9_{\pm 8.2}$	$62.6_{\pm 0.16}$	$47.4_{\pm 0.35}$	**0.60**	**3.65**
	S-MFT Both(Ours)	$58.1_{\pm 0.22}$	$32.0_{\pm 0.32}$	$87.7_{\pm 0.47}$	$1218.0_{\pm 7.7}$	$61.0_{\pm 0.19}$	$48.7_{\pm 0.39}$	0.74	4.32
Gemma-2B	Zero-Shot	2.6	26.4	56.6	586.2	2.8	5.6	-	-
	FFT	$60.9_{\pm 0.37}$	$31.3_{\pm 0.35}$	$86.4_{\pm 0.44}$	$1339.0_{\pm 10.8}$	$60.5_{\pm 0.27}$	$50.7_{\pm 0.26}$	0.88	13.98
	LoRA	$58.3_{\pm 0.32}$	$29.6_{\pm 0.30}$	$86.2_{\pm 0.46}$	$1178.0_{\pm 9.7}$	$57.4_{\pm 0.20}$	$48.0_{\pm 0.23}$	0.64	9.43
	DoRA	$59.6_{\pm 0.28}$	$30.8_{\pm 0.32}$	$86.8_{\pm 0.41}$	$1250.4_{\pm 9.5}$	$59.1_{\pm 0.22}$	$49.8_{\pm 0.24}$	0.61	9.85
	QLoRA	$56.7_{\pm 0.36}$	$29.9_{\pm 0.39}$	$85.1_{\pm 0.46}$	$1152.3_{\pm 10.4}$	$55.8_{\pm 0.25}$	$46.1_{\pm 0.33}$	0.63	10.71
	Parallel Adapter	$58.8_{\pm 0.32}$	$30.5_{\pm 0.32}$	$86.4_{\pm 0.46}$	$1218.7_{\pm 10.4}$	$58.0_{\pm 0.22}$	$48.8_{\pm 0.25}$	0.72	9.75
	P-Tuning v2	$54.6_{\pm 0.29}$	$27.8_{\pm 0.32}$	$82.5_{\pm 0.43}$	$1114.7_{\pm 10.9}$	$53.4_{\pm 0.20}$	$45.9_{\pm 0.25}$	0.82	8.50
	S-MFT Attn (Ours)	$60.5_{\pm 0.24}$	$31.2_{\pm 0.30}$	$87.4_{\pm 0.45}$	$1348.8_{\pm 8.6}$	$63.5_{\pm 0.24}$	$50.3_{\pm 0.21}$	**0.53**	**7.21**
	S-MFT MLP (Ours)	$61.4_{\pm 0.26}$	$31.9_{\pm 0.31}$	$87.4_{\pm 0.47}$	$1402.6_{\pm 9.8}$	$62.4_{\pm 0.22}$	$51.1_{\pm 0.21}$	**0.57**	**8.06**
	S-MFT Both (Ours)	$61.6_{\pm 0.25}$	$31.9_{\pm 0.28}$	$87.6_{\pm 0.49}$	$1361.8_{\pm 9.1}$	$61.2_{\pm 0.20}$	$52.3_{\pm 0.22}$	0.58	8.16
Phi-2-2.7B	Zero-Shot	2.3	25.7	50.1	87.4	9.8	1.5	-	-
	FFT	$60.5_{\pm 0.37}$	$38.2_{\pm 0.32}$	$86.4_{\pm 0.45}$	$1413.9_{\pm 9.6}$	$71.9_{\pm 0.29}$	$56.5_{\pm 0.22}$	0.86	13.53
	LoRA	$59.8_{\pm 0.34}$	$37.1_{\pm 0.22}$	$88.1_{\pm 0.49}$	$1382.3_{\pm 9.7}$	$71.1_{\pm 0.20}$	$53.7_{\pm 0.25}$	0.67	8.15
	DoRA	$61.0_{\pm 0.31}$	$37.9_{\pm 0.26}$	$88.2_{\pm 0.41}$	$1395.2_{\pm 8.8}$	$71.5_{\pm 0.23}$	$55.2_{\pm 0.27}$	0.65	8.65
	QLoRA	$58.3_{\pm 0.39}$	$33.9_{\pm 0.28}$	$85.7_{\pm 0.47}$	$1354.6_{\pm 10.3}$	$69.4_{\pm 0.28}$	$56.9_{\pm 0.37}$	0.72	11.18
	Parallel Adapter	$60.3_{\pm 0.34}$	$36.8_{\pm 0.20}$	$86.8_{\pm 0.45}$	$1388.7_{\pm 9.7}$	$70.8_{\pm 0.27}$	$54.4_{\pm 0.24}$	0.69	8.42
	P-Tuning v2	$57.0_{\pm 0.33}$	$31.5_{\pm 0.39}$	$84.5_{\pm 0.47}$	$1280.6_{\pm 9.6}$	$66.3_{\pm 0.28}$	$51.0_{\pm 0.31}$	0.81	7.84
	S-MFT Attn(Ours)	$60.3_{\pm 0.24}$	$37.8_{\pm 0.27}$	$88.3_{\pm 0.45}$	$1403.9_{\pm 9.3}$	$72.4_{\pm 0.20}$	$56.1_{\pm 0.25}$	**0.46**	**5.21**
	S-MFT MLP (Ours)	$61.4_{\pm 0.25}$	$38.6_{\pm 0.25}$	$88.0_{\pm 0.47}$	$1468.3_{\pm 9.1}$	$71.7_{\pm 0.23}$	$58.8_{\pm 0.31}$	**0.54**	**7.33**
	S-MFT Both(Ours)	$61.9_{\pm 0.26}$	$39.1_{\pm 0.28}$	$88.8_{\pm 0.45}$	$1444.5_{\pm 9.0}$	$72.4_{\pm 0.21}$	$57.2_{\pm 0.28}$	0.64	8.12

Hence, the standard Rademacher generalization bound (Bartlett & Mendelson, 2002) gives MFT a no-larger complexity term at a matched effective-weight budget. This establishes MFT's capacity-control advantage: its structure controls statistical complexity while the experiments show that it preserves or improves downstream task performance. The compression analysis below strengthens this ordering with an explicit sufficient condition for a strictly tighter MFT bound. The full risk statement and proof are in Appendix C.1.

Eq. 8 establishes that MFT's complexity is upper-bounded by FFT's. We next derive an explicit sufficient condition for a strict separation when a learned soft mask M admits a short code. Let $\widetilde{M}$ leave all but s mask coordinates equal to one and encode each selected coordinate with a fixed, data-independent b-bit alphabet. Define its worst-case loss distortion as

$$\alpha(M, \widetilde{M}) = \sup_z \left| \ell(f(\cdot; W_0 \odot M), z) - \ell(f(\cdot; W_0 \odot \widetilde{M}), z) \right|. \tag{9}$$

Since there are at most $\binom{d}{s} 2^{bs}$ such codes, a finite-class bound gives, with probability at least $1 - \delta$,

$$R(h_M) \le \widehat{R}_{\mathcal{S}}(h_M) + 2\alpha(M, \widetilde{M}) + \sqrt{\frac{\ln \binom{d}{s} + bs \ln 2 + \ln(1/\delta)}{2n}}. \tag{10}$$

The analogous numerator for a dense b_F-bit FFT code is $b_F d \ln 2$. Thus, MFT has a strictly smaller complexity contribution whenever $\ln \binom{d}{s} + bs \ln 2 < b_F d \ln 2$; combining this coding advantage with preserved empirical fit and small mask distortion yields a strictly tighter complete risk bound

Table 2: Comparison of MFT and strong mask selection baselines.

	Method	Evaluation Tasks					
		GQA ↑	MMMU ↑	POPE ↑	MME ↑	SQA-Image ↑	TextVQA ↑
Qwen2.5-0.5B	Zero-Shot	5.7	22.6	49.9	93.7	3.2	9.0
	FFT	57.8	32.8	86.4	1278.5	58.2	47.6
	Wanda 0.05%	58.2	32.4	86.1	1290.5	58.8	47.5
	SparseGPT 0.05%	58.0	32.2	85.9	1286.2	58.6	47.4
	H-MFT 0.05%	**59.1**	**33.8**	**86.6**	**1345.7**	**59.4**	**48.0**
	H-MFT 0.01%	56.1	31.7	85.9	1302.1	57.9	43.9
	H-MFT 0.1%	53.0	31.4	85.3	1212.4	57.9	44.8
	S-MFT	59.0	34.6	87.2	1342.1	59.5	48.8
TinyLLaMA-1.1B	Zero-Shot	1.0	26.1	49.8	211.8	4.0	6.8
	FFT	56.9	29.8	86.2	1165.3	59.1	47.2
	Wanda 1.0%	56.5	30.5	86.4	1175.2	58.8	47.6
	SparseGPT 1.0%	56.3	30.3	86.2	1171.5	58.6	47.4
	H-MFT 1.0%	**57.5**	**31.6**	**86.9**	**1191.9**	**59.5**	**48.2**
	H-MFT 0.5%	56.2	28.1	86.7	1174.7	54.9	43.1
	H-MFT 1.5%	51.2	28.2	84.1	1112.4	57.1	46.2
	S-MFT	57.8	32.3	86.8	1196.1	60.7	48.4
Gemma2-2B	Zero-Shot	2.6	26.4	56.6	586.2	2.8	5.6
	FFT	60.2	30.8	86.8	1318.4	59.7	50.2
	Wanda 3%	60.6	30.4	86.4	1325.6	59.8	50.6
	SparseGPT 3%	60.4	30.2	86.3	1322.1	59.5	50.4
	H-MFT 3%	**61.5**	**31.8**	**86.9**	**1351.5**	**60.5**	**51.4**
	H-MFT 1%	55.1	31.4	85.9	1272.3	59.0	47.2
	H-MFT 5%	55.4	30.7	85.5	1242.1	60.0	48.9
	S-MFT	61.4	32.1	87.1	1362.7	60.4	52.0
Phi-2-2.7B	Zero-Shot	2.3	25.7	50.1	87.4	9.8	1.5
	FFT	61.2	37.5	87.2	1398.6	71.4	56.1
	Wanda 0.1%	61.6	37.3	87.1	1408.2	71.6	56.0
	SparseGPT 0.1%	61.4	37.1	87.0	1404.5	71.4	55.8
	H-MFT 0.1%	**62.1**	**38.4**	**87.7**	**1427.0**	**72.1**	**56.6**
	H-MFT 0.05%	61.1	37.1	87.3	1393.3	71.0	56.7
	H-MFT 0.3%	60.7	35.9	86.8	1345.6	69.8	55.4
	S-MFT	62.1	38.8	88.0	1423.5	72.6	57.0

for MFT. The result provides an explicit, empirically checkable mechanism through which structural selection improves generalization over dense FFT updates.

For the score-to-weight map $\Phi(S) = W_0 \odot \sigma_T(S)$,

$$\|\Phi(S_1) - \Phi(S_2)\|_2 \leq \frac{\|W_0\|_\infty}{4T}\|S_1 - S_2\|_2. \tag{11}$$

Moreover, the chain rule gives

$$\begin{aligned} \nabla_S \mathcal{L} &= (\nabla_{W'}\mathcal{L} \odot W_0) \odot \sigma_T'(S), \\ \|\nabla_S \mathcal{L}\|_2 &\leq \frac{\|W_0\|_\infty}{4T}\|\nabla_{W'}\mathcal{L}\|_2 \end{aligned} . \tag{12}$$

These inequalities explicitly bound score-to-weight sensitivity. The sigmoid derivative selectively damps score updates on saturated coordinates, providing a stable mechanism for consolidating learned structural selections. The complete assumptions, post-selection corrections, H-MFT and FFT special cases, and all proofs are in Appendix C.1.

4 Experiments

We apply our Mask Fine-tuning (MFT) method on Vision-Language Models (VLMs) across different vision and language backbones and evaluate on diverse vision-language tasks. We also provide ablation studies and analyses from various perspectives to gain deeper insights into our method.

4.1 Experimental Setup

Vision Encoders. We follow the mainstream VLM training paradigm, in which the vision encoder is frozen, and the projector and language model are trained (Liu et al., 2023b; Zhou et al., 2024). Accordingly, we adopt CLIP (Radford et al., 2021) and SigLIP (Zhai et al., 2023) in our experiments.

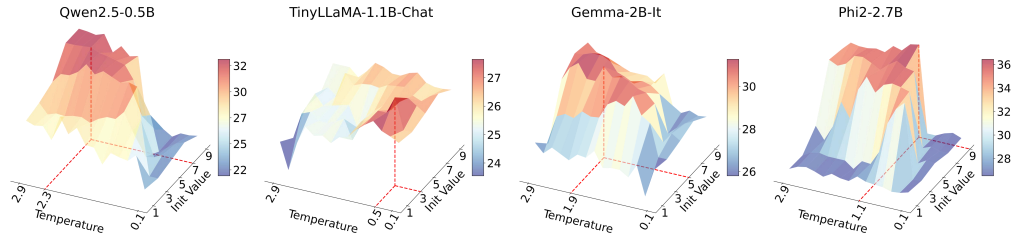

Figure 3: Hyperparameter sensitivity of S-MFT with respect to the initial score value (S_{init}) and temperature (T) across four language backbones. Each surface plot shows the MMMU performance under different hyperparameter combinations. The red dashed line indicates the configuration selected for full training.

Table 3: Ablation of different vision towers.

Method	GQA ↑	MMMU ↑	POPE ↑	MME ↑	SQA-Image ↑	TextVQA ↑
Qwen2.5-0.5B + CLIP-ViT-Large						
FFT	55.7	34.2	85.3	1197.9	57.6	44.2
LoRA	55.1	32.7	85.2	1196.7	57.5	43.7
S-MFT	**56.8**	**35.2**	**86.3**	**1238.4**	**58.9**	**45.2**
TinyLlama-1.1B + CLIP-ViT-Large						
FFT	58.0	27.9	85.5	1284.6	59.9	46.3
LoRA	57.8	27.5	85.4	1278.4	59.3	45.9
S-MFT	**59.2**	**29.0**	**86.5**	**1319.0**	**61.0**	**47.2**
Gemma-2B + CLIP-ViT-Large						
FFT	56.4	32.3	84.4	1194.1	60.0	45.8
LoRA	57.2	28.4	83.9	1249.7	57.6	45.9
S-MFT	**58.4**	**33.4**	**85.8**	**1254.4**	**60.9**	**47.0**
Phi-2-2.7B + CLIP-ViT-Large						
FFT	59.4	36.3	86.8	1448.6	71.2	53.4
LoRA	58.7	35.8	86.6	1437.1	70.9	53.3
S-MFT	**60.3**	**37.3**	**87.9**	**1477.9**	**72.1**	**55.0**

Table 4: Scaling study of S-MFT on VLMs.

Method	GQA ↑	MMMU ↑	POPE ↑	MME ↑	SQA-Image ↑	TextVQA ↑
Qwen2.5-7B + SigLIP-ViT-Large						
FFT	64.7	43.8	87.7	1558.3	76.4	64.0
LoRA	63.9	42.6	86.8	1512.4	74.5	63.7
S-MFT	**65.6**	**44.7**	**88.6**	**1588.1**	**77.7**	**65.3**
Qwen2.5-14B + SigLIP-ViT-Large						
FFT	65.6	47.8	88.5	1597.2	80.1	66.1
LoRA	65.3	46.5	87.8	1568.9	79.1	66.0
S-MFT	**66.5**	**48.9**	**89.6**	**1627.5**	**81.2**	**67.2**
LLaMA2-7B + CLIP-ViT-Large						
FFT	61.7	40.2	87.2	1506.6	71.8	57.7
LoRA	61.9	38.8	87.6	1490.5	71.3	56.5
S-MFT	**62.7**	**41.3**	**88.3**	**1535.9**	**72.9**	**58.7**
LLaMA2-13B + CLIP-ViT-Large						
FFT	63.5	43.6	88.0	1545.2	74.4	59.8
LoRA	63.1	42.4	87.9	1528.8	73.8	59.0
S-MFT	**64.4**	**44.7**	**88.9**	**1578.4**	**75.5**	**61.0**

Language Models. We primarily explore and validate MFT on four capable but compact language backbones: Qwen2.5-0.5B (Qwen et al., 2025), TinyLLaMA-1.1B (Zhang et al., 2024), Gemma-2B (Team et al., 2024), and Phi-2-2.7B (Gunasekar et al., 2023). We further validate the transferability of the discovered structural selection principles through targeted scale-up experiments on larger mainstream language backbones.

Baselines & Datasets. We follow the settings of LLaVA (Liu et al., 2023a) and TinyLLaVA (Zhou et al., 2024), which serve as our strong fine-tuning baselines. Specifically, we use exactly the same training data as in their setups and evaluate our models across multiple downstream tasks, including GQA (Ainslie et al., 2023), MMMU (Yue et al., 2024), POPE (Li et al., 2023), MME (Fu et al., 2025), SQA (Lu et al., 2022), and TextVQA (Singh et al., 2019). We also compare MFT against several representative PEFT approaches, including LoRA (Hu et al., 2021), DoRA (Liu et al., 2024), QLoRA (Dettmers et al., 2023), Parallel Adapter (He et al., 2022), and P-Tuning v2 (Liu et al., 2021), to demonstrate its effectiveness. Additional comparisons are presented in the Appendix.

4.2 Main Results

We use Soft Mask Fine-Tuning (S-MFT) as the primary experimental setting and conduct extensive ablation studies to demonstrate its effectiveness, efficiency, and guidance to the binary variant.

Comparison. Tab. 1 contains our main experiment results of S-MFT, which achieves strong and consistent performance across all four backbones and six evaluation tasks. All experiments follow the TinyLLaVA (Zhou et al., 2024) setting and train for one epoch with SigLIP (Zhai et al., 2023) as the vision tower. We implement three S-MFT variants: 1) S-MFT Attn, 2) S-MFT MLP, and 3) S-MFT Both, which apply learnable masks to the attention weights, MLP weights, or both within Transformer (Wolf et al., 2020) blocks. The "Epochs" and "Time" columns report the training duration required for convergence. S-MFT consistently achieves strictly lower training costs than FFT. Within this reduced time budget, the strongest S-MFT configuration consistently surpasses FFT and PEFT methods across diverse visual-linguistic benchmarks. Among S-MFT variants, S-MFT Both consistently yields the highest overall accuracy, while module-specific masking provides efficient operating points with further computational savings. These results establish S-MFT as a robust and efficient fine-tuning method that surpasses full tuning across diverse VLM architectures.

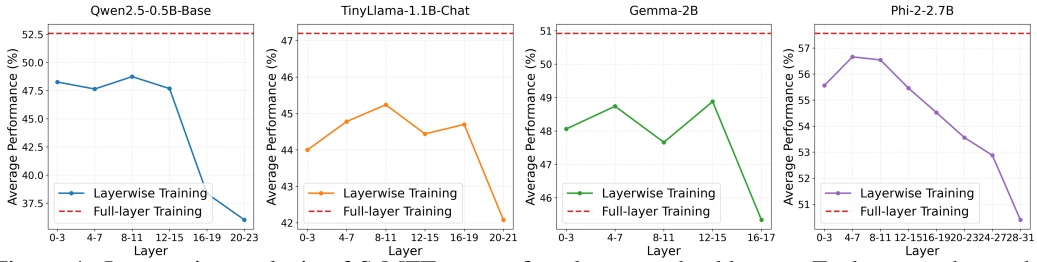

Figure 4: Layer-wise analysis of S-MFT across four language backbones. Each curve shows the average performance when masks are applied only to a subset of layers, compared with full-layer training (red dashed line).

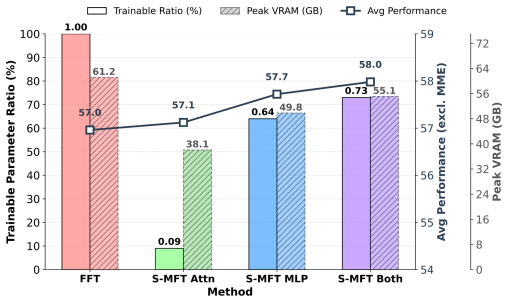

Figure 5: Comparison of training cost and average performance between FFT and MFT.

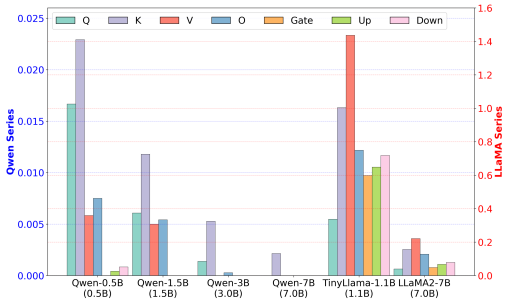

Figure 6: The proportion of near-zero mask values learned by S-MFT.

H-MFT vs. Pruning. In Tab. 2, H-MFT achieves the best performance when using the approximated sparsity level adaptively learned by S-MFT (Fig. 6). A key insight of MFT is that adapting multimodal models does not require additional information. Instead, effective adaptation can be achieved by modifying only a tiny fraction of connections via mask selection. To verify that the performance gains arise from task-aware structural selection rather than generic sparsity effects, we compare H-MFT with Wanda (Sun et al., 2023) and SparseGPT (Frantar & Alistarh, 2023) at identical sparsities. To ensure strict optimization parity, we first derive static pruning masks for the base model and then perform sparse fine-tuning under the same multimodal training configuration as H-MFT. These task-agnostic heuristics select weights using magnitude or general activation statistics, providing controls that isolate the contribution of multimodal task supervision. H-MFT's task-aware masks consistently surpass FFT and both heuristic control groups. These results show that MFT's gains arise from learned structural selection rather than generic sparsity or regularization.

4.3 Ablation Studies

Hyperparameters. S-MFT mainly depends on two hyperparameters: the initial score S_{init} and the temperature T. Before full-scale training, we perform a lightweight grid search on 10% of the training data to evaluate each model's sensitivity to these two hyperparameters (Fig. 3). As shown in Fig. 3, the performance surfaces show broad, continuous high-performance regions rather than sharp peaks, indicating that S-MFT is robust to variations in both S_{init} and T. We therefore select the optimal configuration as the default setting for full-dataset training. Concretely, we use (S_{init}-T) values of 7.0-2.3, 3.0-0.5, 5.0-1.9, and 9.0-1.1 for Qwen2.5-0.5B, TinyLLaMA-1.1B, Gemma-2B, and Phi-2-2.7B, respectively.

Vision Towers. We evaluate S-MFT with CLIP (Radford et al., 2021) in addition to the default SigLIP (Zhai et al., 2023) tower to validate cross-encoder generality in Tab. 3. S-MFT consistently outperforms FFT and LoRA across these vision-language combinations, demonstrating robust generalization across vision towers.

Scale-Up. To demonstrate the scalability of our method, we extend S-MFT to larger language backbones in the Qwen2.5 and LLaMA2 families. Models within each family exhibit similar mask distributions (Fig. 6), enabling direct zero-shot transfer of (S_{init}, T) from the corresponding sub-billion-parameter proxy model. Without any additional hyperparameter search on the larger models, S-MFT consistently outperforms FFT and LoRA across the reported downstream benchmarks. These results

establish strong architecture-family transferability and amortize the initial profiling cost, yielding a practical "proxy-tuning" paradigm in which selection priors discovered on a compact model transfer directly to substantially larger models.

Layer-wise Ablation. We conduct a layer-wise sensitivity study with 10% of the training data to examine where mask learning is most effective (Fig. 4). Across all four backbones, focused masking of mid-level layers achieves the highest average performance and approaches full-layer training. This concentration of gains identifies mid-level reconfiguration as the most efficient adaptation region, providing a principled route to further reduce training cost through targeted layer selection.

5 Analysis

Training Memory. Unlike traditional PEFT, designed for extreme parameter reduction at the cost of narrow informational bottlenecks, MFT is fundamentally a high-capacity, structural selection alternative to FFT. During the backward pass, MFT preserves the dense forward activations required to compute score gradients through the chain rule, retaining the model's full cross-modal representational capacity. MFT obtains its memory advantage by freezing the pre-trained weights and eliminating their optimizer states; optimizer storage is confined to trainable scores in the selected masked modules. Empirically, in Fig. 5, on a Qwen2.5-0.5B backbone, FFT consumes 61.2 GB of peak VRAM, while S-MFT Both uses 55.1 GB while exploring a broad 73% trainable score space. S-MFT Attn provides an especially efficient operating point, which optimizes only 9% of the parameter space, reducing peak VRAM to 38.1 GB, avoiding the FFT out-of-memory regime, and preserving dense-level multimodal alignment.

Mask structure. Fig. 6 reports the proportion of learned mask values below 0.1 for each projection. Together with the task-aware pruning controls in Tab. 2, the differentiated, architecture-specific attenuation patterns provide evidence that S-MFT learns structured selections rather than applying uniform shrinkage. These endpoint statistics identify candidate coordinates for the compact approximations analyzed by Eq. 10, providing empirical inputs for evaluating MFT's compression-based generalization advantage.

Loss landscape. Fig. 7 complements the formal analysis with a filter-normalized two-dimensional view of the loss near the learned solutions. The S-MFT endpoint occupies a visibly wider low-loss region than FFT, providing qualitative evidence of stronger local perturbation robustness. Following the visualization procedure of (Li et al., 2018), we use filter-normalized directions to control simple scale effects. The broader S-MFT basin provides complementary empirical evidence that MFT combines the controlled hypothesis complexity established in Sec. 3.4 with stronger local robustness along the displayed directions.

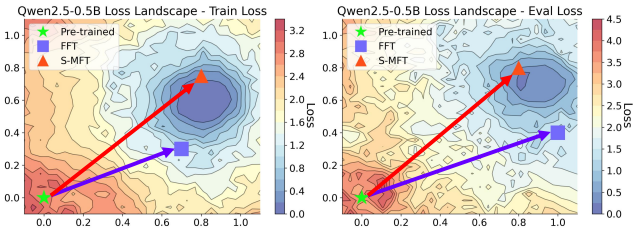

Figure 7: Loss landscape of the Qwen2.5-0.5B backbone.

6 Conclusion

In this paper, we rethink VLM adaptation by shifting from traditional weight-update paradigms to structural selection, introducing Mask Fine-Tuning (MFT). Rather than destructively updating pre-trained weights or injecting additive modules that impose capacity bottlenecks, MFT structurally selects and reorganizes the redundant representational pathways within the frozen backbone. Our theory establishes MFT's controlled statistical complexity and gives explicit sufficient conditions for a strictly tighter generalization bound than dense FFT. Empirically, MFT consistently outperforms FFT and PEFT across architectures, scales, and benchmarks while achieving lower training cost and zero inference overhead. Together, these results establish structural selection as an effective and general VLM adaptation paradigm.

AI USE STATEMENT

We use AI to correct grammar errors and help plot figures.

ETHICS STATEMENT

In our paper, we strictly adhere to the ICLR ethical research standards and applicable laws. To the best of our knowledge, our work abides by the General Ethical Principles.

REPRODUCIBILITY STATEMENT

We adhere to ICLR reproducibility standards and ensure the reproducibility of our work. All datasets we employed are publicly available. We will provide the code to reviewers and area chairs via an anonymous link, if needed, during the discussion period.

REFERENCES

Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Dhruv Batra, and Devi Parikh. Vqa: Visual question answering, 2016.

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebrón, and Sumit Sanghai. Gqa: Training generalized multi-query transformer models from multi-head checkpoints, 2023.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Yue Bai, Huan Wang, ZHIQIANG TAO, Kunpeng Li, and Yun Fu. Dual lottery ticket hypothesis. In *International Conference on Learning Representations*, 2022.

Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of machine learning research*, 2002.

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation, 2013.

Jinhe Bi, Yujun Wang, Haokun Chen, Xun Xiao, Artur Hecker, Volker Tresp, and Yunpu Ma. LLaVA steering: Visual instruction tuning with 500x fewer parameters through modality linear representation-steering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. ACL, 2025.

Florian Bordes, Richard Yuanzhe Pang, Anurag Ajay, Alexander C Li, Adrien Bardes, Suzanne Petryk, Oscar Mañas, Zhiqiu Lin, Anas Mahmoud, Bargav Jayaraman, et al. An introduction to vision-language modeling. *arXiv preprint arXiv:2405.17247*, 2024.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186. Association for Computational Linguistics, June 2019.

Gongfan Fang, Xinyin Ma, and Xinchao Wang. Structural pruning for diffusion models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 16716–16728. Curran Associates, Inc., 2023.

Gongfan Fang, Xinyin Tan Ma, Michael Bi Mi, and Xinchao Wang. Isomorphic pruning for vision models. *arXiv preprint arXiv:2407.04616*, 2024a.

Gongfan Fang, Hongxu Yin, Saurav Muralidharan, Greg Heinrich, Jeff Pool, Jan Kautz, Pavlo Molchanov, and Xinchao Wang. Maskllm: Learnable semi-structured sparsity for large language models. *arXiv preprint arXiv:2409.17481*, 2024b.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks, 2019.

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

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, Rongrong Ji, Caifeng Shan, and Ran He. Mme: A comprehensive evaluation benchmark for multimodal large language models, 2025.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023.

Song Han, Xingyu Liu, Huizi Mao, Jing Pu, Ardavan Pedram, Mark A. Horowitz, and William J. Dally. Eie: efficient inference engine on compressed deep neural network. In *Proceedings of the 43rd International Symposium on Computer Architecture*, ISCA '16, pp. 243–254. IEEE Press, 2016a. ISBN 9781467389471.

Song Han, Huizi Mao, and William J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding, 2016b.

Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. In *International Conference on Learning Representations*, 2022.

Bairu Hou, Qibin Chen, Jianyu Wang, Guoli Yin, Chong Wang, Nan Du, Ruoming Pang, Shiyu Chang, and Tao Lei. Instruction-following pruning for large language models, 2025.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, 2019.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021.

Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and ¡0.5mb model size, 2016.

Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. Ocr-free document understanding transformer. In *European Conference on Computer Vision (ECCV)*, 2022.

Dawid J. Kopiczko, Tijmen Blankevoort, and Yuki M. Asano. Vera: Vector-based random matrix adaptation, 2024.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Fei-Fei Li. Visual genome: Connecting language and vision using crowdsourced dense image annotations, 2016.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3045–3059. Association for Computational Linguistics, November 2021.

Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. In *Advances in Neural Information Processing Systems*, 2018.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022.

Kaiyang Li, Shaobo Han, Qing Su, Wei Li, Zhipeng Cai, and Shihao Ji. Uni-lora: One vector is all you need. In *The 39th Conference on Neural Information Processing Systems (NeurIPS)*, 2025a.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation, 2021.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Zongxia Li, Xiyang Wu, Hongyang Du, Fuxiao Liu, Huy Nghiem, and Guangyao Shi. A survey of state of the art large vision language models: Alignment, benchmark, evaluations and challenges, 2025b.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015.

Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning, 2022a.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023a.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023b.

Hongyi Liu, Rajarshi Saha, Zhen Jia, Youngsuk Park, Jiaji Huang, Shoham Sabach, Yu-Xiang Wang, and George Karypis. Proxsparse: Regularized learning of semi-structured sparsity masks for pretrained llms. *arXiv preprint arXiv:2502.00258*, 2025a.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: Weight-decomposed low-rank adaptation, 2024.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *CoRR*, 2021.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 61–68. Association for Computational Linguistics, May 2022b.

Yiheng Liu, Junhao Ning, Sichen Xia, Xiaohui Gao, Ning Qiang, Bao Ge, Junwei Han, and Xintao Hu. Pruning large language models by identifying and preserving functional networks. *arXiv preprint arXiv:2508.05239*, 2025b.

Yuanxin Liu, Fandong Meng, Zheng Lin, Peng Fu, Yanan Cao, Weiping Wang, and Jie Zhou. Learning to win lottery tickets in BERT transfer via task-agnostic mask training. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5840–5857. Association for Computational Linguistics, July 2022c.

Yuchen Liu, Yaoming Wang, Bowen Shi, Xiaopeng Zhang, Wenrui Dai, Chenglin Li, Hongkai Xiong, and Qi Tian. Meteor: Multi-encoder collaborative token pruning for efficient vision language models, 2025c.

Zhuang Liu, Mingjie Sun, Tinghui Zhou, Gao Huang, and Trevor Darrell. Rethinking the value of network pruning. In *ICLR*, 2019.

Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *The 36th Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

Xinyin Ma, Gongfan Fang, and Xinchao Wang. Llm-pruner: On the structural pruning of large language models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 21702–21720. Curran Associates, Inc., 2023.

Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. Ocr-vqa: Visual question answering by reading text in images. In *ICDAR*, 2019.

Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. Pruning convolutional neural networks for resource efficient inference, 2017.

Changdae Oh, Hyesu Lim, Mijoo Kim, Dongyoon Han, Sangdoo Yun, Jaegul Choo, Alexander Hauptmann, Zhi-Qi Cheng, and Kyungwoo Song. Towards calibrated robust fine-tuning of vision-language models, 2024.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022.

Tan-Hanh Pham and Chris Ngo. Rarl: Improving medical vlm reasoning and generalization with reinforcement learning and lora under data and hardware constraints, 2025.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

Amanpreet Singh, Vivek Natarjan, Meet Shah, Yu Jiang, Xinlei Chen, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8317–8326, 2019.

Mingjie Sun, Zhuang Liu, Anna Bair, and J. Zico Kolter. A simple and effective pruning approach for large language models. *arXiv preprint arXiv:2306.11695*, 2023.

Mingjie Sun, Zhuang Liu, Anna Bair, and J. Zico Kolter. A simple and effective pruning approach for large language models, 2024.

Chaofan Tao, Lu Hou, Haoli Bai, Jiansheng Wei, Xin Jiang, Qun Liu, Ping Luo, and Ngai Wong. Structured pruning for efficient generative pre-trained language models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 10880–10895. Association for Computational Linguistics, July 2023.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023.

Huan Wang and Yun Fu. Trainability preserving neural pruning. In *ICLR*, 2023.

Huan Wang, Can Qin, Yulun Zhang, and Yun Fu. Neural pruning via growing regularization. In *International Conference on Learning Representations (ICLR)*, 2021.

Huan Wang, Can Qin, Yue Bai, Yulun Zhang, and Yun Fu. Recent advances on neural network pruning at initialization, 2022.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45. Association for Computational Linguistics, October 2020.

Yi Xu, Yuxin Hu, Zaiwei Zhang, Gregory P. Meyer, Siva Karthik Mustikovela, Siddhartha Srinivasa, Eric M. Wolff, and Xin Huang. Vlm-ad: End-to-end autonomous driving through vision-language model supervision, 2025.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of CVPR*, 2024.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training, 2023.

Yuexiang Zhai, Hao Bai, Zipeng Lin, Jiayi Pan, Shengbang Tong, Yifei Zhou, Alane Suhr, Saining Xie, Yann LeCun, Yi Ma, and Sergey Levine. Fine-tuning large vision-language models as decision-making agents via reinforcement learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Dan Zhang, Tao Feng, Lilong Xue, Yuandong Wang, Yuxiao Dong, and Jie Tang. Parameter-efficient fine-tuning for foundation models, 2025a.

Mingyuan Zhang, Yue Bai, Huan Wang, Yizhou Wang, Qihua Dong, and Yun Fu. Boosting large language models with mask fine-tuning, 2025b.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. Tinyllama: An open-source small language model, 2024.

Yuanhe Zhang, Fanghui Liu, and Yudong Chen. Lora-one: One-step full gradient could suffice for fine-tuning large language models, provably and efficiently, 2025c.

Mengjie Zhao, Tao Lin, Fei Mi, Martin Jaggi, and Hinrich Schütze. Masking as an efficient alternative to finetuning for pretrained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2226–2241. Association for Computational Linguistics, November 2020.

Kecheng Zheng, Wei Wu, Ruili Feng, Kai Zhu, Jiawei Liu, Deli Zhao, Zheng-Jun Zha, Wei Chen, and Yujun Shen. Regularized mask tuning: Uncovering hidden knowledge in pre-trained vision-language models. In *ICCV*, 2023a.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023b.

Baichuan Zhou, Ying Hu, Xi Weng, Junlong Jia, Jie Luo, Xien Liu, Ji Wu, and Lei Huang. Tinyllava: A framework of small-scale large multimodal models, 2024.

APPENDIX

Table 5: Overview of datasets used across the pretraining, SFT, and evaluation stages. Training set sizes are denoted by the number of images, while evaluation sets are denoted by the number of query instances (QAs).

Dataset Name	Samples	Usage
Pretraining		
LAION-CC-SBU-558K (Liu et al., 2023b)	558,128	Training
Supervised Fine-Tuning (SFT)		
LAION-CC-SBU-558K (Liu et al., 2023b)	558,128	Training
COCO (train2017) (Lin et al., 2015)	118,287	Training
GQA (Ainslie et al., 2023)	113,018	Training
OCR-VQA (Mishra et al., 2019)	207,572	Training
TextVQA (Singh et al., 2019)	28,408	Training
VisualGenome (Krishna et al., 2016)	108,077	Training
Evaluation		
GQA (Ainslie et al., 2023)	12,578	Evaluation
MMMU (Yue et al., 2024)	11,500	Evaluation
POPE (Li et al., 2023)	3,000	Evaluation
MME (Fu et al., 2025)	2,374	Evaluation
SQA (Lu et al., 2022)	4,241	Evaluation
TextVQA (Singh et al., 2019)	5,000	Evaluation

A IMPLEMENTATION DETAILS

A.1 DATASET DETAILS

The training and evaluation datasets in our experiments are identical to those used by LLaVA (Liu et al., 2023b) and TinyLLaVA (Zhou et al., 2024). Tab. 5 summarizes the dataset splits.

A.2 TRAINING DETAILS

Our experiments are conducted on a server equipped with 4 NVIDIA A100 GPUs with 80GB of memory. We borrow the code of LLaVA (Liu et al., 2023b) and TinyLLaVA (Zhou et al., 2024), as well as their LLaVA-v1.5-Mix-665K instruction dataset for projector pre-training and the BLIP-Laion-CC-SBU-558K instruction dataset for fine-tuning. We also use the official evaluation protocols for both codebases in all our experiments.

B EXPERIMENTS

B.1 COMPLETE MAIN RESULTS

We present our main results in Tab. 6, including additional fine-tuning methods (LoRA-One (Zhang et al., 2025c), Uni-LoRA (Li et al., 2025a), Adapter (Houlsby et al., 2019), MoReS (Bi et al., 2025), IA^3 (Liu et al., 2022a)). The results demonstrate that our S-MFT consistently outperforms all other methods, aligning with our claim in the main text.

B.2 LEARNING RATE ABLATION

S-MFT initializes every mask entry close to one, so the initial effective weights are close to the pre-trained weights:

$$W \odot M_{\text{init}} \approx W \odot \mathbf{1} = W, \tag{13}$$

where $\mathbf{1}$ denotes the all-ones matrix. The exact initial mask is $\sigma(S_{\text{init}}/T)$ and is strictly below one for finite S_{init}. Because the score gradient is multiplied coordinate-wise by $W \odot \sigma_T'(S)$ in Eq. 12,

Table 6: Main experimental result comparison with the time cost of our Soft Mask Fine-Tuning (S-MFT) with Full Fine-Tuning (FFT) and representative Parameter-Efficient Fine-Tuning (PEFT) baselines across four language backbones. We repeat each experiment five times with different random seeds and report the mean and standard deviation. Red numbers denote the best results, and blue numbers denote the second-best. All results reported here are the performance after a full training epoch. "Epochs" and "Time" are reported as the efficiency metric, indicating when a method converges and achieves stable, optimal validation performance.

	Method	Evaluation Tasks						Epoch ↓	Time (h) ↓
		GQA ↑	MMMU ↑	POPE ↑	MME ↑	SQA-Image ↑	TextVQA ↑		
Qwen2.5-0.5B	Zero-Shot	5.7	22.6	49.9	93.7	3.2	9.0	-	-
	FFT	$58.3_{\pm 0.28}$	$33.3_{\pm 0.37}$	$86.1_{\pm 0.46}$	$1253.0_{\pm 10.2}$	$58.9_{\pm 0.25}$	$48.2_{\pm 0.23}$	0.90	4.91
	LoRA	$55.6_{\pm 0.28}$	$32.0_{\pm 0.35}$	$86.9_{\pm 0.45}$	$1206.0_{\pm 10.1}$	$58.6_{\pm 0.22}$	$45.5_{\pm 0.24}$	0.65	3.14
	DoRA	$57.1_{\pm 0.24}$	$32.8_{\pm 0.31}$	**87.1**$_{\pm 0.42}$	$1228.5_{\pm 9.4}$	$58.8_{\pm 0.21}$	$46.8_{\pm 0.28}$	0.63	3.42
	QLoRA	$53.8_{\pm 0.36}$	$30.1_{\pm 0.41}$	$85.2_{\pm 0.47}$	$1178.4_{\pm 10.2}$	$56.9_{\pm 0.27}$	$46.7_{\pm 0.39}$	0.88	5.97
	LoRA-One	$56.0_{\pm 0.25}$	$29.9_{\pm 0.32}$	$87.0_{\pm 0.49}$	$1208.4_{\pm 9.5}$	$57.2_{\pm 0.20}$	$45.5_{\pm 0.23}$	0.88	4.14
	Uni-LoRA	$52.5_{\pm 0.21}$	$32.0_{\pm 0.34}$	$86.0_{\pm 0.46}$	$1227.3_{\pm 10.9}$	$58.0_{\pm 0.29}$	$42.4_{\pm 0.39}$	0.69	3.98
	Adapter	$56.4_{\pm 0.26}$	$31.3_{\pm 0.35}$	$82.5_{\pm 0.44}$	$1284.8_{\pm 10.5}$	$56.2_{\pm 0.23}$	$44.4_{\pm 0.31}$	0.75	4.02
	Parallel Adapter	$56.0_{\pm 0.25}$	$30.8_{\pm 0.32}$	$86.5_{\pm 0.49}$	$1215.4_{\pm 9.5}$	$57.6_{\pm 0.20}$	$45.8_{\pm 0.23}$	0.72	3.95
	P-Tuning v2	$52.8_{\pm 0.21}$	$28.5_{\pm 0.34}$	$84.1_{\pm 0.46}$	$1162.3_{\pm 10.9}$	$54.2_{\pm 0.29}$	$42.4_{\pm 0.39}$	0.85	3.58
	MoReS	$57.7_{\pm 0.24}$	$32.1_{\pm 0.33}$	$84.8_{\pm 0.41}$	**1310.5**$_{\pm 9.2}$	$57.8_{\pm 0.26}$	$46.3_{\pm 0.29}$	0.78	4.15
	IA³	$55.3_{\pm 0.31}$	$30.5_{\pm 0.42}$	$81.3_{\pm 0.48}$	$1259.8_{\pm 11.2}$	$55.1_{\pm 0.28}$	$43.2_{\pm 0.35}$	0.68	3.15
	S-MFT Attn(Ours)	$58.0_{\pm 0.27}$	$33.3_{\pm 0.31}$	$86.7_{\pm 0.47}$	$1243.0_{\pm 8.2}$	**59.7**$_{\pm 0.20}$	$47.9_{\pm 0.37}$	**0.61**	**2.92**
	S-MFT MLP (Ours)	**59.0**$_{\pm 0.21}$	**33.4**$_{\pm 0.32}$	**87.3**$_{\pm 0.45}$	$1306.0_{\pm 8.4}$	**60.0**$_{\pm 0.17}$	**48.9**$_{\pm 0.39}$	**0.62**	**3.13**
	S-MFT Both(Ours)	**59.5**$_{\pm 0.22}$	**34.8**$_{\pm 0.34}$	$87.0_{\pm 0.48}$	**1349.3**$_{\pm 8.6}$	$59.2_{\pm 0.19}$	**49.4**$_{\pm 0.38}$	0.67	3.81
TinyLlama-1.1B	Zero-Shot	1.0	26.1	49.8	211.8	4.0	6.8	-	-
	FFT	$57.3_{\pm 0.33}$	$30.4_{\pm 0.34}$	$85.8_{\pm 0.44}$	$1180.9_{\pm 9.2}$	$60.3_{\pm 0.27}$	$47.9_{\pm 0.23}$	0.93	5.76
	LoRA	$57.5_{\pm 0.41}$	$30.4_{\pm 0.22}$	$86.9_{\pm 0.47}$	$1167.6_{\pm 9.5}$	$58.2_{\pm 0.19}$	$47.9_{\pm 0.27}$	0.77	4.53
	DoRA	$57.4_{\pm 0.31}$	**30.9**$_{\pm 0.25}$	$87.1_{\pm 0.43}$	$1175.2_{\pm 9.8}$	$59.8_{\pm 0.18}$	$48.1_{\pm 0.26}$	0.72	4.81
	QLoRA	$55.9_{\pm 0.35}$	$26.2_{\pm 0.35}$	$81.5_{\pm 0.48}$	$1142.8_{\pm 11.5}$	$56.5_{\pm 0.22}$	$46.2_{\pm 0.38}$	0.62	5.03
	LoRA-One	$57.2_{\pm 0.30}$	$29.4_{\pm 0.31}$	$87.2_{\pm 0.48}$	$1170.5_{\pm 8.9}$	$59.5_{\pm 0.21}$	$46.2_{\pm 0.24}$	0.66	4.15
	Uni-LoRA	$55.4_{\pm 0.27}$	$27.1_{\pm 0.28}$	$84.5_{\pm 0.44}$	$1122.0_{\pm 11.8}$	$54.7_{\pm 0.14}$	$44.4_{\pm 0.22}$	0.63	3.92
	Adapter	$56.6_{\pm 0.32}$	$29.5_{\pm 0.29}$	$86.2_{\pm 0.45}$	$1163.4_{\pm 9.6}$	$57.3_{\pm 0.25}$	$46.5_{\pm 0.27}$	0.84	4.68
	Parallel Adapter	$56.8_{\pm 0.30}$	$29.8_{\pm 0.31}$	$86.4_{\pm 0.48}$	$1165.5_{\pm 8.9}$	$57.5_{\pm 0.21}$	$46.8_{\pm 0.24}$	0.81	4.62
	P-Tuning v2	$53.4_{\pm 0.27}$	$25.1_{\pm 0.28}$	$82.5_{\pm 0.44}$	$1105.0_{\pm 11.8}$	$52.7_{\pm 0.14}$	$44.4_{\pm 0.22}$	0.88	4.25
	MoReS	$57.7_{\pm 0.38}$	$30.6_{\pm 0.24}$	$87.3_{\pm 0.44}$	$1185.6_{\pm 8.8}$	$59.2_{\pm 0.17}$	$48.0_{\pm 0.25}$	0.75	4.85
	IA³	$54.2_{\pm 0.31}$	$26.5_{\pm 0.35}$	$83.1_{\pm 0.51}$	$1118.2_{\pm 10.2}$	$53.8_{\pm 0.19}$	$45.1_{\pm 0.32}$	0.68	3.75
	S-MFT Attn(Ours)	$57.1_{\pm 0.20}$	$30.7_{\pm 0.29}$	$87.6_{\pm 0.47}$	$1170.9_{\pm 7.4}$	$60.4_{\pm 0.15}$	**49.1**$_{\pm 0.37}$	**0.61**	**2.98**
	S-MFT MLP (Ours)	**57.9**$_{\pm 0.30}$	**30.9**$_{\pm 0.20}$	**88.0**$_{\pm 0.48}$	**1237.9**$_{\pm 8.2}$	**62.6**$_{\pm 0.16}$	$47.4_{\pm 0.35}$	**0.60**	**3.65**
	S-MFT Both(Ours)	**58.1**$_{\pm 0.22}$	**32.0**$_{\pm 0.32}$	**87.7**$_{\pm 0.47}$	**1218.0**$_{\pm 7.7}$	**61.0**$_{\pm 0.19}$	**48.7**$_{\pm 0.39}$	0.74	4.32
Gemma-2B	Zero-Shot	2.6	26.4	56.6	586.2	2.8	5.6	-	-
	FFT	$60.9_{\pm 0.37}$	**31.3**$_{\pm 0.35}$	$86.4_{\pm 0.44}$	$1339.0_{\pm 10.8}$	$60.5_{\pm 0.27}$	$50.7_{\pm 0.26}$	0.88	13.98
	LoRA	$58.3_{\pm 0.32}$	$29.6_{\pm 0.30}$	$86.2_{\pm 0.46}$	$1178.0_{\pm 9.7}$	$57.4_{\pm 0.20}$	$48.0_{\pm 0.23}$	0.64	9.43
	DoRA	$59.6_{\pm 0.28}$	$30.8_{\pm 0.32}$	$86.8_{\pm 0.41}$	$1250.4_{\pm 9.5}$	$59.1_{\pm 0.22}$	$49.8_{\pm 0.24}$	0.61	9.85
	QLoRA	$56.7_{\pm 0.36}$	$29.9_{\pm 0.39}$	$85.1_{\pm 0.46}$	$1152.3_{\pm 10.4}$	$55.8_{\pm 0.25}$	$46.1_{\pm 0.33}$	0.63	10.71
	LoRA-One	$58.4_{\pm 0.32}$	$30.1_{\pm 0.32}$	$86.0_{\pm 0.46}$	$1224.7_{\pm 10.4}$	$58.4_{\pm 0.22}$	$49.5_{\pm 0.25}$	0.70	8.19
	Uni-LoRA	$56.6_{\pm 0.29}$	$30.4_{\pm 0.32}$	$83.5_{\pm 0.43}$	$1194.7_{\pm 10.9}$	$57.4_{\pm 0.20}$	$49.9_{\pm 0.25}$	0.71	9.10
	Adapter	$58.6_{\pm 0.33}$	$30.3_{\pm 0.35}$	$86.1_{\pm 0.48}$	$1215.3_{\pm 10.1}$	$57.8_{\pm 0.25}$	$48.6_{\pm 0.28}$	0.75	9.92
	Parallel Adapter	$58.8_{\pm 0.32}$	$30.5_{\pm 0.32}$	$86.4_{\pm 0.46}$	$1218.7_{\pm 10.4}$	$58.0_{\pm 0.22}$	$48.8_{\pm 0.25}$	0.72	9.75
	P-Tuning v2	$54.6_{\pm 0.29}$	$27.8_{\pm 0.32}$	$82.5_{\pm 0.43}$	$1114.7_{\pm 10.9}$	$53.4_{\pm 0.20}$	$45.9_{\pm 0.25}$	0.82	8.50
	MoReS	$59.8_{\pm 0.29}$	$31.0_{\pm 0.31}$	$87.0_{\pm 0.42}$	$1265.5_{\pm 9.3}$	$59.6_{\pm 0.21}$	$50.1_{\pm 0.26}$	0.65	10.15
	IA³	$55.4_{\pm 0.35}$	$28.5_{\pm 0.38}$	$83.6_{\pm 0.47}$	$1135.8_{\pm 11.2}$	$54.2_{\pm 0.24}$	$46.5_{\pm 0.31}$	0.71	8.15
	S-MFT Attn (Ours)	$60.5_{\pm 0.24}$	$31.2_{\pm 0.30}$	**87.4**$_{\pm 0.45}$	$1348.8_{\pm 8.6}$	**63.5**$_{\pm 0.24}$	$50.3_{\pm 0.21}$	**0.53**	**7.21**
	S-MFT MLP (Ours)	**61.4**$_{\pm 0.26}$	**31.9**$_{\pm 0.31}$	**87.4**$_{\pm 0.47}$	**1402.6**$_{\pm 9.8}$	**62.4**$_{\pm 0.22}$	**51.1**$_{\pm 0.21}$	**0.57**	**8.06**
	S-MFT Both (Ours)	**61.6**$_{\pm 0.25}$	**31.9**$_{\pm 0.28}$	**87.6**$_{\pm 0.49}$	**1361.8**$_{\pm 9.1}$	$61.2_{\pm 0.20}$	**52.3**$_{\pm 0.22}$	0.58	8.16
Phi-2-2.7B	Zero-Shot	2.3	25.7	50.1	87.4	9.8	1.5	-	-
	FFT	$60.5_{\pm 0.37}$	$38.2_{\pm 0.32}$	$86.4_{\pm 0.45}$	$1413.9_{\pm 9.6}$	**71.9**$_{\pm 0.29}$	$56.5_{\pm 0.22}$	0.86	13.53
	LoRA	$59.8_{\pm 0.34}$	$37.1_{\pm 0.22}$	$88.1_{\pm 0.49}$	$1382.3_{\pm 9.7}$	$71.1_{\pm 0.20}$	$53.7_{\pm 0.25}$	0.67	8.15
	DoRA	$61.0_{\pm 0.31}$	$37.9_{\pm 0.26}$	$88.2_{\pm 0.41}$	$1395.2_{\pm 8.8}$	$71.5_{\pm 0.23}$	$55.2_{\pm 0.27}$	0.65	8.65
	QLoRA	$58.3_{\pm 0.39}$	$33.9_{\pm 0.28}$	$85.7_{\pm 0.47}$	$1354.6_{\pm 10.3}$	$69.4_{\pm 0.28}$	$56.9_{\pm 0.37}$	0.72	11.18
	LoRA-One	$60.1_{\pm 0.34}$	$35.9_{\pm 0.20}$	$85.4_{\pm 0.45}$	$1380.7_{\pm 9.7}$	$69.6_{\pm 0.27}$	$53.0_{\pm 0.24}$	0.60	7.69
	Uni-LoRA	$59.0_{\pm 0.33}$	$35.0_{\pm 0.39}$	$86.5_{\pm 0.47}$	$1370.6_{\pm 9.6}$	$70.3_{\pm 0.28}$	**58.0**$_{\pm 0.31}$	0.72	9.94
	Adapter	$60.1_{\pm 0.36}$	$36.5_{\pm 0.22}$	$86.6_{\pm 0.48}$	$1385.2_{\pm 10.1}$	$70.5_{\pm 0.29}$	$54.1_{\pm 0.26}$	0.71	8.55
	Parallel Adapter	$60.3_{\pm 0.34}$	$36.8_{\pm 0.20}$	$86.8_{\pm 0.45}$	$1388.7_{\pm 9.7}$	$70.8_{\pm 0.27}$	$54.4_{\pm 0.24}$	0.69	8.42
	P-Tuning v2	$57.0_{\pm 0.33}$	$31.5_{\pm 0.39}$	$84.5_{\pm 0.47}$	$1280.6_{\pm 9.6}$	$66.3_{\pm 0.28}$	$51.0_{\pm 0.31}$	0.81	7.84
	MoReS	$61.2_{\pm 0.30}$	$38.0_{\pm 0.24}$	$88.2_{\pm 0.43}$	$1410.5_{\pm 9.2}$	$71.6_{\pm 0.22}$	$55.8_{\pm 0.29}$	0.66	8.85
	IA³	$57.8_{\pm 0.37}$	$32.5_{\pm 0.41}$	$85.1_{\pm 0.50}$	$1305.4_{\pm 10.8}$	$67.5_{\pm 0.32}$	$51.8_{\pm 0.35}$	0.75	7.35
	S-MFT Attn(Ours)	$60.3_{\pm 0.24}$	$37.8_{\pm 0.27}$	**88.3**$_{\pm 0.45}$	$1403.9_{\pm 9.3}$	**72.4**$_{\pm 0.20}$	$56.1_{\pm 0.25}$	**0.46**	**5.21**
	S-MFT MLP (Ours)	**61.4**$_{\pm 0.25}$	**38.6**$_{\pm 0.25}$	$88.0_{\pm 0.47}$	**1468.3**$_{\pm 9.1}$	$71.7_{\pm 0.23}$	**58.8**$_{\pm 0.31}$	**0.54**	**7.33**
	S-MFT Both(Ours)	**61.9**$_{\pm 0.26}$	**39.1**$_{\pm 0.28}$	**88.8**$_{\pm 0.45}$	**1444.5**$_{\pm 9.0}$	**72.4**$_{\pm 0.21}$	$57.2_{\pm 0.28}$	0.64	8.12

initialization and temperature directly affect the scale of score updates. We therefore report the learning-rate ablation in Fig. 9; hyperparameter selection must use a held-out validation split that is disjoint from every reported evaluation set.

B.3 Data Efficiency.

We further evaluate S-MFT's data efficiency by varying the fraction of training data used during fine-tuning (Fig. 8). Across all four backbones, S-MFT consistently outperforms FFT and LoRA under different data ratios. The advantage becomes particularly pronounced in low-data regimes, suggesting that the structural selection learned by S-MFT enables more efficient adaptation when supervision is limited.

C Analysis

C.1 Mathematical Proofs

This appendix proves the claims in Sec. 3.4. The probability statements are conditional on the fixed pre-trained weights W_0 and assume i.i.d. downstream examples and a per-example loss in $[0, 1]$. All mask and FFT quantization alphabets and all candidate compression configurations used by the compression bounds are fixed independently of the downstream training sample.

Technical setup and full statements. Let $\mathcal{S} = \{z_i\}_{i=1}^n$ be drawn i.i.d. from $\mathcal{D}$, and define

$$R(h) = \mathbb{E}_{z\sim\mathcal{D}}[\ell(h, z)], \qquad \widehat{R}_{\mathcal{S}}(h) = \tfrac{1}{n}\textstyle\sum_{i=1}^n \ell(h, z_i), \tag{14}$$

where $\ell(h, z) \in [0, 1]$ is a bounded evaluation loss or a pre-specified clipped and rescaled training loss. The pre-trained weights W_0 are treated as fixed before $\mathcal{S}$ is observed. The matched comparison in Sec. 3.4 uses the same d mask-eligible coordinates and the same effective update budget for MFT and FFT, isolating the statistical benefit of structural selection.

For a radius r fixed independently of $\mathcal{S}$, applying the standard empirical Rademacher bound to the two classes in Eq. 7 and taking a union bound gives, with probability at least $1 - \delta$, simultaneously for $A \in \{\mathrm{MFT}, \mathrm{FFT}\}$ and all $h \in \mathcal{H}_A(r)$,

$$R(h) \le \widehat{R}_{\mathcal{S}}(h) + 2\widehat{\mathfrak{R}}_n(\ell \circ \mathcal{H}_A(r)) + 3\sqrt{\frac{\ln(4/\delta)}{2n}}. \tag{15}$$

Together with Eq. 8, this establishes the no-larger complexity term claimed in the main text. If r is chosen after observing $\mathcal{S}$ from a finite grid $\mathcal{R}$ fixed in advance, the same argument replaces $\ln(4/\delta)$ by $\ln(4|\mathcal{R}|/\delta)$.

Compression codebook and special cases. For fixed integers $s \le d$ and $b \ge 1$, let $\mathcal{C}_{s,b}$ contain masks that select s coordinate slots, set every unselected value to one, and assign each selected coordinate one of at most 2^b values from a quantization alphabet fixed before observing $\mathcal{S}$. Then

$$|\mathcal{C}_{s,b}| \le \binom{d}{s} 2^{bs}. \tag{16}$$

The bound in Eq. 10 is stated for fixed s, b, and alphabet. If (s, b) is selected using $\mathcal{S}$ from a finite candidate set $\mathcal{K}$ declared in advance, the numerator acquires an additional $\ln|\mathcal{K}|$ term. If the loss is G-Lipschitz in the effective weights on the relevant parameter set, the distortion in the main text satisfies

$$\alpha(M, \widetilde{M}) \le G\|W_0 \odot (M - \widetilde{M})\|_2. \tag{17}$$

For H-MFT with exactly s zeros and all other mask entries equal to one, value quantization is unnecessary and $\alpha = 0$. Thus,

$$R(h_{\text{H-MFT}}) \le \widehat{R}_{\mathcal{S}}(h_{\text{H-MFT}}) + \sqrt{\tfrac{\ln\binom{d}{s} + \ln(1/\delta)}{2n}}. \tag{18}$$

The combinatorial term already accounts for selecting the mask from the training sample. Selecting s from a predeclared finite candidate set requires the same additional $\ln|\mathcal{K}|$ term.

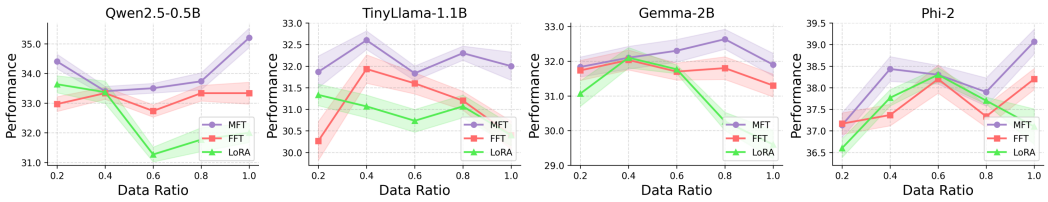

Figure 8: Performance comparison of S-MFT, FFT, and LoRA under varying training data ratios across four language backbones. Each curve shows the MMMU benchmark performance, with shaded regions indicating the standard deviation. S-MFT consistently outperforms FFT and LoRA across varying data ratios, with particularly pronounced advantages in low-data regimes.

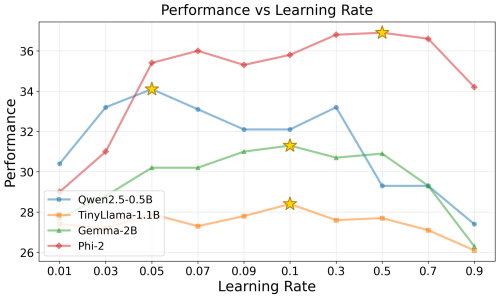

Figure 9: Performance across four language backbones on the MMMU with different learning rates.

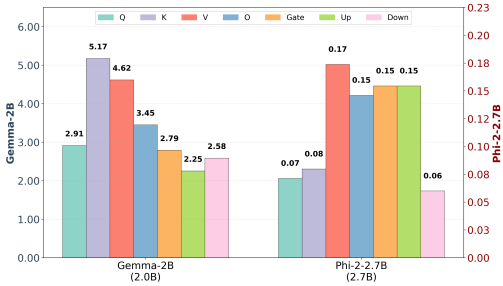

Figure 10: Learned mask distributions for two language backbones, Gemma-2B and Phi-2-2.7B.

For comparison, fix a dense b_F-bit, coordinate-wise code for FFT, with at most $2^{b_F d}$ codewords. For a coded approximation $\widetilde{W}_F$ to the learned FFT weights, let $\alpha_F = \sup_z |\ell(h_F, z) - \ell(h_{\widetilde{W}_F}, z)|$. The same finite-class argument gives

$$R(h_F) \le \widehat{R}_{\mathcal{S}}(h_F) + 2\alpha_F + \sqrt{\frac{b_F d \ln 2 + \ln(1/\delta)}{2n}}. \tag{19}$$

If b_F or its alphabet is selected using $\mathcal{S}$, the corresponding candidate-set term is included. The strict comparison evaluates the complete right-hand sides of Eqs. 10 and 19 using the bounded empirical losses, compression distortions, and stated code lengths. This protocol directly quantifies when MFT's shorter structural code translates into a tighter generalization guarantee.

Proof of Lemma. *The MFT reparameterization function $\Phi(S) = W_0 \odot \sigma_T(S)$ is globally Lipschitz continuous with constant $L_\Phi = \frac{\|W_0\|_\infty}{4T}$.*

Proof. Let $S \in \mathbb{R}^d$. The Jacobian matrix of the transformation $\Phi(S)$ is a diagonal matrix $J_\Phi(S) \in \mathbb{R}^{d\times d}$, where the i-th diagonal element is given by:

$$[J_\Phi(S)]_{ii} = \frac{\partial \Phi_i}{\partial S_i} = W_{0,i} \cdot \tfrac{1}{T}\sigma_T(S_i)\big(1 - \sigma_T(S_i)\big). \tag{20}$$

The derivative of the standard sigmoid function $\sigma(z)$ attains its global maximum of $1/4$ at $z = 0$. Therefore, for the temperature-scaled sigmoid, we have $\max_{S_i} \frac{\partial \sigma_T(S_i)}{\partial S_i} = \frac{1}{4T}$.

To find the Lipschitz constant L_Φ, we bound the spectral norm (maximum singular value) of the Jacobian matrix:

$$L_\Phi = \sup_S \|J_\Phi(S)\|_2 = \sup_S \max_i \left|W_{0,i} \cdot \tfrac{1}{4T}\right| = \tfrac{\|W_0\|_\infty}{4T}. \tag{21}$$

For any $S_1, S_2 \in \mathbb{R}^d$, define the line segment $\gamma(t) = S_2 + t(S_1 - S_2)$ for $t \in [0, 1]$. The fundamental theorem of calculus applied along this path gives

$$\Phi(S_1) - \Phi(S_2) = \int_0^1 J_\Phi(\gamma(t))(S_1 - S_2)\, dt. \tag{22}$$

Taking Euclidean norms and using the uniform Jacobian bound yields

$$\begin{aligned}\|\Phi(S_1) - \Phi(S_2)\|_2 &\le \int_0^1 \|J_\Phi(\gamma(t))\|_2 \, dt \, \|S_1 - S_2\|_2 \\ &\le \frac{\|W_0\|_\infty}{4T} \|S_1 - S_2\|_2\end{aligned}. \tag{23}$$

which proves Eq. 11.

Gradient-sensitivity consequence. For any differentiable empirical objective $\mathcal{L}(W')$, the multivariate chain rule and the diagonal form of $J_\Phi(S)$ give

$$\nabla_S \mathcal{L} = J_\Phi(S)^\top \nabla_{W'} \mathcal{L} = (\nabla_{W'} \mathcal{L} \odot W_0) \odot \sigma_T'(S). \tag{24}$$

Submultiplicativity of the Euclidean operator norm and the preceding lemma yield

$$\|\nabla_S \mathcal{L}\|_2 \le \|J_\Phi(S)\|_2 \|\nabla_{W'} \mathcal{L}\|_2 \le \tfrac{\|W_0\|_\infty}{4T} \|\nabla_{W'} \mathcal{L}\|_2. \tag{25}$$

This proves Eq. 12 and explicitly bounds the score-space sensitivity contributed by the MFT reparameterization relative to the effective-weight gradient.

Proof of the update-class inclusion. For any $M \in [0,1]^d$, write

$$\Delta W_M = W_0 \odot (M - \mathbf{1}). \tag{26}$$

Coordinate-wise, $M_i - 1 \in [-1, 0]$, and hence $\Delta W_{M,i} W_{0,i} = W_{0,i}^2 (M_i - 1) \le 0$ and $|\Delta W_{M,i}| \le |W_{0,i}|$. If additionally $\|\Delta W_M\|_2 \le r$, then ΔW_M belongs to the Euclidean ball $\mathcal{U}_{\mathrm{FFT}}(r)$ by definition. Therefore $\mathcal{U}_{\mathrm{MFT}}(r) \subseteq \mathcal{U}_{\mathrm{FFT}}(r)$. Applying the same map $\Delta W \mapsto [x \mapsto f(x; W_0 + \Delta W)]$ to both sets proves $\mathcal{H}_{\mathrm{MFT}}(r) \subseteq \mathcal{H}_{\mathrm{FFT}}(r)$.

For a fixed sample, empirical Rademacher complexity is

$$\widehat{\mathfrak{R}}_n(\mathcal{G}) = \mathbb{E}_\xi \left[\sup_{g \in \mathcal{G}} \frac{1}{n} \sum_{i=1}^n \xi_i g(z_i) \right], \tag{27}$$

where the ξ_i are independent Rademacher variables. Since a supremum over a subset cannot exceed the supremum over its superset, applying this definition to the two loss classes proves Eq. 8. For each fixed loss class, the standard empirical Rademacher theorem for $[0,1]$-valued functions gives the corresponding inequality in Eq. 15 with failure probability $\delta/2$. A union bound over the two classes replaces $\ln(2/(\delta/2))$ by $\ln(4/\delta)$ and proves that Eq. 15 holds simultaneously for MFT and FFT. If the radius is selected from a predeclared finite grid $\mathcal{R}$, allocating failure probability $\delta/(2|\mathcal{R}|)$ to each method–radius pair gives the stated $\ln(4|\mathcal{R}|/\delta)$ post-selection term.

Proof of the compression bound. Let $\mathcal{C}_{s,b}$ be the fixed coded-mask class from Eq. 16, and write $N_{s,b} = |\mathcal{C}_{s,b}|$. For any fixed $\widetilde{M} \in \mathcal{C}_{s,b}$, Hoeffding's inequality for the bounded loss gives

$$\Pr\left[R(h_{\widetilde{M}}) - \widehat{R}_{\mathcal{S}}(h_{\widetilde{M}}) > t \right] \le \exp(-2nt^2). \tag{28}$$

Taking a union bound over all $N_{s,b}$ coded masks and setting

$$t = \sqrt{\frac{\ln N_{s,b} + \ln(1/\delta)}{2n}} \tag{29}$$

shows that, with probability at least $1 - \delta$, simultaneously for all $\widetilde{M} \in \mathcal{C}_{s,b}$,

$$R(h_{\widetilde{M}}) \le \widehat{R}_{\mathcal{S}}(h_{\widetilde{M}}) + \sqrt{\frac{\ln N_{s,b} + \ln(1/\delta)}{2n}}. \tag{30}$$

This event is uniform over the codebook, so it remains valid when the training sample selects the coded mask. If (s, b) is itself selected from a predeclared finite set $\mathcal{K}$, another union bound allocates failure probability $\delta/|\mathcal{K}|$ to each configuration and adds $\ln |\mathcal{K}|$ to the numerator. This proves the post-selection variant stated after Eq. 10.

Now let M be any soft mask and $\widetilde{M} \in \mathcal{C}_{s,b}$ any coded approximation. By the definition of $\alpha(M, \widetilde{M})$,

$$R(h_M) \le R(h_{\widetilde{M}}) + \alpha(M, \widetilde{M}), \tag{31}$$

$$\widehat{R}_{\mathcal{S}}(h_{\widetilde{M}}) \le \widehat{R}_{\mathcal{S}}(h_M) + \alpha(M, \widetilde{M}). \tag{32}$$

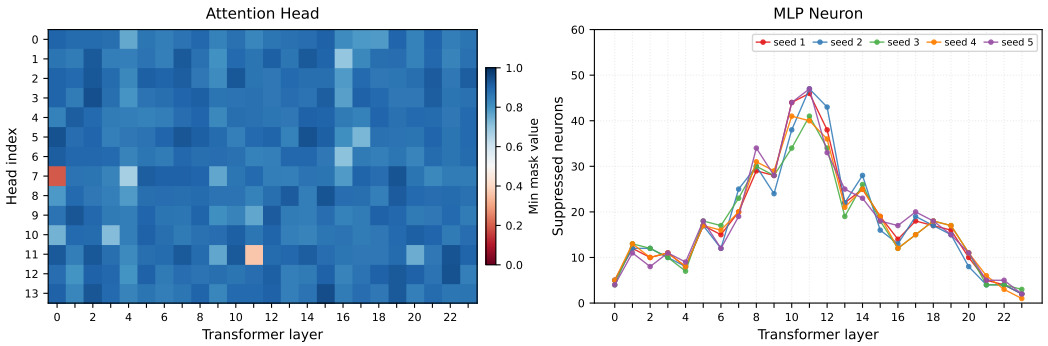

Figure 11: Analysis of block-level masks across 5 seeds.

Combining these two inequalities with Eq. 30 and using $N_{s,b} \leq \binom{d}{s} 2^{bs}$ proves Eq. 10.

If $\ell(f(\cdot; W), z)$ is G-Lipschitz in W uniformly over z on the line segments joining the original and coded effective weights, then

$$\begin{aligned} \alpha(M, \widetilde{M}) &\leq G \|W_0 \odot M - W_0 \odot \widetilde{M}\|_2 \\ &= G \|W_0 \odot (M - \widetilde{M})\|_2, \end{aligned} \tag{33}$$

which proves Eq. 17.

For an H-MFT mask with exactly s zeros, the selected coordinates and their fixed zero values completely determine the mask. There are $\binom{d}{s}$ such masks, and choosing $\widetilde{M} = M$ makes $\alpha = 0$. Substitution in Eq. 30 proves Eq. 18.

Finally, a fixed dense b_F-bit FFT code has at most $2^{b_F d}$ codewords. Repeating the same uniform Hoeffding argument with its loss distortion α_F proves Eq. 19. Comparing the complete right-hand sides gives the stated sufficient criterion: MFT's shorter structural code, together with preserved empirical fit and small approximation distortion, yields a strictly tighter bound than dense FFT.

C.2 Additional Mask Analysis

Fig. 10 shows that Gemma-2B has a larger proportion of mask values below 0.1 than Phi-2-2.7B in the displayed projections. These architecture-specific attenuation signatures show that S-MFT adapts its structural selection to each backbone's organization. Targeted intervention studies that remove, restore, or exchange selected connections provide a promising direction for further characterizing the causal roles of these pathways.

C.3 Block-level masks.

We use weight-level masks, given their flexibility, to explore the effectiveness of structural selection. We add Block-level S-MFT on LLaVA-Qwen2.5-0.5B for comparison, which learns masks on Attn heads and MLP neurons. We directly transfer the best hyperparameters from the weight-level setting to make an initial attempt, evaluated on 4 tasks.

Table 7: Mask granularity comparison on LLaVA-Qwen2.5-0.5B.

Method	MMMU ↑	POPE ↑	SQA ↑	TextVQA ↑
Block	31.8	86.5	58.7	43.7
Weight	**35.1**	**87.5**	**59.4**	**49.6**

Because block-level masks are less flexible than weight-level masks, their lower performance is expected. But the moderate gap suggests it may be improved with dedicated hyperparameter tuning. We further ran both methods with the same hyperparameters across 5 seeds. Weight-level masks are stable in overall sparsity, not in individual weight positions, even though performance is similar. In contrast, block-level masks show stable unit-level patterns. One Attn unit (layer 0, head 7) is always strongly suppressed, while most others remain active. Redundancy is mainly observed in MLP neurons, with about 400 neurons suppressed per run, concentrated in the middle layers (especially layers 10-12, in Fig. 11). These reproducible unit-level patterns provide a more interpretable view

of where redundancy is concentrated. Understanding why these specific units are selected requires a deeper mechanistic analysis, which is beyond the scope of this submission. We believe this is a meaningful research direction, as block-level masks may also be more hardware-friendly.

C.4 Inference & Deployment.

The learned masks are merged offline by multiplying them directly into the frozen weights ($W' = W \odot M$), guaranteeing zero inference overhead. H-MFT additionally supports lightweight task-specific binary masks alongside a shared backbone, substantially reducing multi-task storage compared with maintaining separate fine-tuned models or adapters.

C.5 Expressivity through Pretrained-Structure Preservation.

S-MFT parameterizes a targeted effective-weight family that preserves each pre-trained connection's sign while learning its task-specific contribution through continuous attenuation. This geometry produces the class inclusion in Eq. 7 and the resulting statistical-complexity advantage. Across all evaluated tasks, this pretrained, structure-preserving family consistently outperforms solutions obtained by FFT and PEFT. The architecture-specific patterns in Figs. 6 and 10 further show that MFT deploys its capacity selectively, adapting the location and strength of attenuation to each backbone.