WANDA-GUIDED SPARSE PERTURBATIONS FOR EFFICIENT ZEROTH-ORDER OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Zeroth-order (ZO) optimization reduces the memory cost of LLM fine-tuning by replacing backpropagation with forward-only gradient estimation. Its practical efficiency deteriorates in high-dimensional parameter spaces because the variance of random-direction estimators increases with the search dimension. We introduce Wanda-Importance Sparsity-MeZO (WIS-MeZO), which restricts perturbations to coordinates selected using both weight magnitude and downstream activation statistics. This support is constructed without gradients or pre-training data, and the pretrained model remains fully dense throughout fine-tuning. To analyze the effect of support selection, we use gradient energy capture to measure how much of the local optimization signal lies inside the perturbed coordinates. Under explicit conditions, WIS-selected supports retain more gradient energy than random sparse supports, and the resulting convergence bound depends on both the effective search dimension and the captured gradient energy. On 16 benchmarks with LLaMA and Qwen models ranging from 3B to 32B parameters, WIS-MeZO consistently outperforms MeZO and random sparse perturbations in aggregate performance, remains competitive with or improves upon magnitude-based support selection, and reaches comparable validation loss in fewer optimization steps.

1 INTRODUCTION

The memory cost of LLM fine-tuning is largely tied to backpropagation, which requires retaining activations and maintaining gradients and optimizer states throughout training. Parameter-efficient fine-tuning (Hu et al., 2022; Liu et al., 2024), quantization (Dettmers et al., 2022), and memory-efficient optimizers (Zhao et al., 2024; Zhu et al., 2025) reduce different parts of this footprint, but do not remove the need for backpropagation itself. Zeroth-order (ZO) optimization provides an alternative by estimating gradients from forward evaluations. MeZO (Malladi et al., 2023), for example, reconstructs perturbations from random seeds and avoids storing backpropagation states.

A central limitation of ZO optimization is its dependence on the search dimension. As more parameters are perturbed, random-direction gradient estimates become noisier. Existing methods mitigate this variance using more function evaluations (Dang et al., 2026) or more informative perturbation distributions based on curvature or preconditioning (Zhao et al., 2025b;a). Sparse perturbations instead reduce the effective search dimension directly, shifting the key question to which coordinates should be perturbed. SensZOQ (Guo et al., 2025) selects supports using pre-training gradients, which may be unavailable, whereas Sparse-MeZO (Liu et al., 2025) relies on task-agnostic weight magnitudes. Yet weight magnitude alone does not capture downstream feature activation and is therefore only an indirect proxy for the task-specific optimization signal.

We propose Wanda-Importance Sparsity-MeZO (WIS-MeZO), which uses Wanda-style weight–activation saliency (Sun et al., 2024) to select sparse ZO perturbation supports while leaving the pretrained model dense. We further analyze support quality through gradient energy capture and connect it to ZO estimation and convergence.

Our main contributions are:

- **Task-Conditioned Sparse Support.** We propose WIS-MeZO, a sparse ZO fine-tuning method that selects perturbation supports using downstream weight–activation saliency.

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.

Unlike random or magnitude-only support selection, WIS-MeZO incorporates task-dependent activation information when allocating the perturbation budget, while leaving the model fully dense and requiring neither gradients nor access to pre-training data.

- **Gradient-Energy Analysis of Support Quality.** We study sparse ZO optimization from the perspective of how much gradient energy is captured by the selected coordinates. Our analysis connects WIS scores to expected gradient energy under explicit conditions and derives a convergence guarantee that depends jointly on the effective search dimension and the captured gradient energy. Comparisons with random, small-magnitude, and large-magnitude supports further clarify why support selection matters, when magnitude-based strategies can be effective, and where they become insufficient.
- **Evaluation Across Tasks, Models, and Scales.** We evaluate WIS-MeZO on 16 benchmarks using LLaMA and Qwen models ranging from 3B to 32B parameters. WIS-MeZO consistently improves aggregate performance over MeZO and random sparse perturbations, remains competitive with or stronger than magnitude-based support selection, and reaches comparable validation loss in fewer optimization steps.

2 Related Work

Zeroth-Order Optimization. ZO optimization estimates gradients through function evaluations rather than backpropagation (Spall, 2002; Ilyas et al., 2018). Classical variants include ZO-SGD (Ghadimi & Lan, 2013), and ZO-Adam (Chen et al., 2019), while MeZO (Malladi et al., 2023) enables memory-efficient LLM fine-tuning through seed-replayed perturbations. Subsequent work improves variance, perturbation geometry, or search dimensionality through variance reduction (Gautam et al., 2024), low-rank or structured subspaces (Zhang et al., 2025; Yu et al., 2025; Chen et al., 2025), preconditioning (Zhao et al., 2025a), and additional queries (Dang et al., 2026). AGZO (Lin et al., 2026) further constructs activation-derived low-rank subspaces to improve gradient alignment. In contrast, WIS-MeZO uses downstream activations to select coordinate-wise perturbation supports and characterizes their quality through gradient energy capture.

Sparse Optimization. Pruning methods such as Wanda (Sun et al., 2024) and SparseGPT (Frantar & Alistarh, 2023) use activation or curvature information to identify removable weights, while dynamic sparse training updates sparsity patterns during first-order optimization (Evci et al., 2020; Joo et al., 2026). Sparse ZO methods instead restrict perturbation coordinates to reduce the effective estimator dimension (Ohta et al., 2020; Gu et al., 2021; Cai et al., 2022; Chen et al., 2023; Zhang et al., 2024; Guo et al., 2025; Liu et al., 2025). DeepZero (Chen et al., 2023) uses finite-difference information, SensZOQ (Guo et al., 2025) relies on pre-training sensitivities, and Sparse-MeZO (Liu et al., 2025) selects coordinates using weight magnitudes. WIS-MeZO instead constructs task-conditioned coordinate supports from downstream activations while keeping the underlying model dense.

3 Preliminaries

In this section, we present the common formulations of ZO optimization typically employed for LLM training (Malladi et al., 2023) and followed by the Wanda importance score (Sun et al., 2024).

3.1 Zeroth-order Optimization

Zeroth-order optimization approximates gradients by only forward passes. Following MeZO (Malladi et al., 2023), for an objective function $f(\mathbf{W})$, a mini-batch $\mathcal{B}$ and a weight matrix $\mathbf{W} \in \mathbb{R}^{m\times n}$, where m denotes the d_{out} and n denotes d_{in}, the stochastic gradient $\mathbf{G} = \nabla f(\mathbf{W}, \mathcal{B})$ is estimated via a finite-difference approximation along a random perturbation $\mathbf{Z} \in \mathbb{R}^{m\times n}$ with the same shape as $\mathbf{W}$. Given a perturbation scale $\epsilon > 0$, the standard ZO gradient estimator is defined as

$$\hat{\mathbf{G}} = \frac{f(\mathbf{W} + \epsilon\mathbf{Z}, \mathcal{B}) - f(\mathbf{W} - \epsilon\mathbf{Z}, \mathcal{B})}{2\epsilon}\mathbf{Z},$$

where the entries of $\mathbf{Z}$ are typically sampled independently from a standard Gaussian or Rademacher distribution. In the limit $\epsilon \to 0$, the expectation of the estimator satisfies $\mathbb{E}_{\mathbf{Z}}[\hat{\mathbf{G}}] = \nabla f(\mathbf{W}, \mathcal{B})$, providing an unbiased direction for optimization without backpropagation.

Utilizing the estimated gradient, we can update the weight by the following recursion

$$\mathbf{W}^{t+1} = \mathbf{W}^t - \eta \cdot \hat{\mathbf{G}}^t,$$

where η denotes the step size.

3.2 WANDA IMPORTANCE SCORE

Wanda (Sun et al., 2024) identifies salient weights by considering both weight magnitudes and input feature scales. Consider the forward pass $\mathbf{Y} = \mathbf{W}\mathbf{X}$, where $\mathbf{X} \in \mathbb{R}^{n \times B}$ contains B calibration examples. For a randomly drawn downstream example, let $\boldsymbol{\delta}_i := \partial f / \partial \mathbf{Y}_i$ denote the backpropagated error signal and define the per-example gradient contribution as $\mathbf{G}_{i,j} := \boldsymbol{\delta}_i \mathbf{X}_j$. The proxy analysis below concerns its population energy $\mathbb{E}[\mathbf{G}_{i,j}^2]$, which avoids imposing additional assumptions on cross-example terms in a mini-batch gradient. Our empirical diagnostic separately evaluates the resulting supports using the actual mini-batch gradients produced during training. The Wanda Importance Score (WIS) for weight $\mathbf{W}_{i,j}$ is defined as

$$\mathbf{S}_{i,j} := |\mathbf{W}_{i,j}| \cdot \|\mathbf{X}_j\|_2, \tag{1}$$

where $\|\mathbf{X}_j\|_2 = \sqrt{\sum_{b=1}^{B} (\mathbf{X}_j^{(b)})^2}$, $|\cdot|$ denotes the absolute value operator, and $\|\cdot\|_2$ represents the ℓ_2 norm. To analyze the relationship between $\mathbf{S}_{i,j}$ and gradient energy, we define the Empirical Second Moment of the j-th input feature as

$$\hat{\mathbf{V}}_j := \frac{1}{B} \sum_{b=1}^{B} (\mathbf{X}_j^{(b)})^2, \quad \mathbf{V}_j = \mathbb{E}[(\mathbf{X}_j^{(b)})^2], \tag{2}$$

such that the squared Wanda score relates to the feature variance via $\mathbf{S}_{i,j}^2 = B\mathbf{W}_{i,j}^2 \hat{\mathbf{V}}_j$. As $\mathbf{V}_j$ captures the intensity of input activations, it serves as a crucial proxy for the expected squared gradient $\mathbb{E}[\mathbf{G}_{i,j}^2]$, which we utilize in our convergence analysis.

4 METHODOLOGY

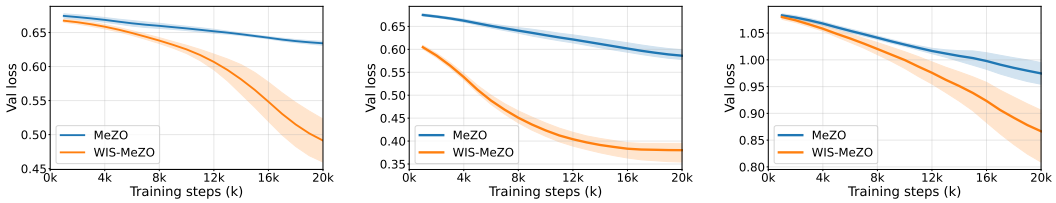
This section introduces Wanda-Importance Sparsity-MeZO (WIS-MeZO), a novel zeroth-order optimization method that leverages the Wanda importance score to guide sparse perturbations. By focusing perturbations on the most salient parameters, WIS-MeZO significantly reduces the effective search dimensionality while preserving the gradient signal.

4.1 WIS-GUIDED SPARSE PERTURBATION

A fundamental challenge in ZO optimization for LLMs is the "curse of dimensionality," where the variance of the gradient estimator scales linearly with the number of parameters $d = mn$. WIS-MeZO addresses this by restricting the perturbation to a sparse subset of parameters. Specifically, for each layer $\mathbf{W} \in \mathbb{R}^{m \times n}$, we define a binary mask $\mathbf{M} \in \{0,1\}^{m \times n}$ based on the Wanda scores $\mathbf{S}_{i,j}$. Given a target row-wise density $r = k/n$, where k is the number of selected parameters per row (resulting in a sparsity ratio of $1 - r$), the mask $\mathbf{M}_{i,j}$ is set to 1 if $\mathbf{S}_{i,j}$ is among the top-k scores in the i-th row, and 0 otherwise. The perturbation direction is constructed as $\frac{1}{\sqrt{r}}(\mathbf{Z} \odot \mathbf{M})$, where $\mathbf{Z} \in \mathbb{R}^{m \times n}$ has i.i.d. standard Gaussian entries. This $\frac{1}{\sqrt{r}}$ scaling ensures that the expected energy of the sparse perturbation remains consistent with a full-space update, while the effective dimension is reduced from mn to mk. Following previous work (Sun et al., 2024), we implement sparsity via a row-wise mask, ensuring that each output neuron maintains a consistent proportion of non-zero input connections. This approach also prevents degenerate cases where certain rows might be entirely pruned or fully preserved.

4.2 EFFICIENT FEATURE VARIANCE ESTIMATION

To compute the Wanda scores $\mathbf{S}_{i,j} = |\mathbf{W}_{i,j}| \cdot \sqrt{B\hat{\mathbf{V}}_j}$ with high precision, WIS-MeZO performs an initial calibration forward pass on a representative mini-batch to estimate the empirical second



(a) RTE (b) BoolQ (c) MNLI

Figure 1: **Validation loss versus training steps for fine-tuning LLaMA3.2-3B, averaged over five random seeds.** WIS-MeZO demonstrates faster convergence than MeZO.

moment $\hat{\mathbf{V}}_j = \frac{1}{B}\sum_{b=1}^{B}(\mathbf{X}_j^{(b)})^2$. This calibration batch size B is typically chosen to be larger than the subsequent training batches to ensure a stable estimation of feature variances (Sun et al., 2024; Frantar et al., 2023). For fixed-mask WIS-MeZO, this calibration is performed once and amortized over the subsequent optimization steps.

For Transformer architectures (Vaswani et al., 2017), $\hat{\mathbf{V}}_j$ corresponds to the intensity of specific input features, such as hidden states. In LLMs, these features often exhibit a "heavy-tailed" distribution where a few dimensions (outliers) possess significantly higher variance, a phenomenon widely documented as a key characteristic of LLM (Dettmers et al., 2022; Xiao et al., 2023). By incorporating $\hat{\mathbf{V}}_j$, WIS-MeZO prioritizes parameters associated with these high-variance features, which carry the largest gradient components and dominate the optimization trajectory.

4.3 The WIS-MeZO Algorithm

In each iteration t, we sample a single random seed to generate the perturbation $\mathbf{Z}^t$. The gradient estimator is computed using two forward passes (or one when using the single-query variant):

$$\hat{\mathbf{G}}^t = \frac{f\left(\mathbf{W}^t + \epsilon\frac{\mathbf{Z}^t}{\sqrt{r}} \odot \mathbf{M}, \mathcal{B}\right) - f\left(\mathbf{W}^t - \epsilon\frac{\mathbf{Z}^t}{\sqrt{r}} \odot \mathbf{M}, \mathcal{B}\right)}{2\epsilon} \cdot \frac{\mathbf{Z}^t}{\sqrt{r}} \odot \mathbf{M}.$$

Here, $\odot$ denotes element-wise multiplication. The weights are updated as $\mathbf{W}^{t+1} = \mathbf{W}^t - \eta\hat{\mathbf{G}}^t$. Because $\mathbf{M}$ is sparse, the update affects only mk parameters. The normalization $\mathbf{Z}^t/\sqrt{r}$ preserves the expected perturbation energy of vanilla MeZO. WIS-MeZO computes $\mathbf{M}$ once and keeps it fixed, whereas WISD-MeZO optionally refreshes it every τ steps. The fixed-mask variant is used as our primary method. Algorithm 1 presents the main optimization procedure. The implementations of $\mathrm{PerturbParameters}$ and $\mathrm{ComputeMask}$ are provided in Appendix D. Although Algorithm 1 writes the ZO gradient estimator explicitly for clarity, the implementation regenerates the perturbation from the same random seed and applies the update layer by layer in place, without storing an additional perturbed model copy.

Mask Generation and Update Strategy For WIS-MeZO, the mask is computed once at the onset of training and cached throughout optimization. For a linear layer $\mathbf{W} \in \mathbb{R}^{m\times n}$, computing the feature statistics over B calibration samples costs $\mathcal{O}(Bn)$, followed by row-wise top-k selection. WISD-MeZO optionally recomputes the mask every τ steps.

Memory and Storage Efficiency Because perturbations are applied in place with seed replay, the main persistent overhead relative to MeZO is the binary mask over the selected linear modules. The default Boolean representation requires 1 byte per masked parameter. For larger models, WIS-MeZO (1-bit) stores the mask in a bit-packed format, reducing this cost to 1 bit per parameter; for a 32B model, the mask then occupies less than 4GB.

5 Theoretical Analysis

Sparse ZO methods are often motivated by reducing the effective search dimension and thereby the variance of gradient estimation. This perspective, however, does not distinguish between different

Algorithm 1 WIS-MeZO

Input: Initial weights $\mathbf{W}^0$, learning rate η, perturbation scale ϵ, mask update frequency τ, density ratio r, calibration dataset $\mathcal{D}_{\text{cal}}$
for step $t = 0, 1, 2, \ldots, T-1$ **do**
  if $t \bmod \tau = 0$ **then**
    Fetch activations $\mathbf{X} \sim \mathcal{D}_{\text{cal}}$ {Collect calibration activations}
    $\mathbf{M} \leftarrow \text{ComputeMask}(\mathbf{W}^t, \mathbf{X}, r)$ {Update mask using Wanda scores}
  end if
  Sample training batch $\mathcal{B}$ and random seed s
  $\mathbf{W}^t \leftarrow \text{PerturbParameters}(\mathbf{W}^t, r, \epsilon, \mathbf{M}, s)$
  $L_+ \leftarrow f(\mathbf{W}^t, \mathcal{B})$
  $\mathbf{W}^t \leftarrow \text{PerturbParameters}(\mathbf{W}^t, r, -2\epsilon, \mathbf{M}, s)$ {Move to the negative perturbation}
  $L_- \leftarrow f(\mathbf{W}^t, \mathcal{B})$
  $\mathbf{W}^t \leftarrow \text{PerturbParameters}(\mathbf{W}^t, r, \epsilon, \mathbf{M}, s)$ {Recover parameters}
  Reset the random number generator with seed s and regenerate $\mathbf{Z}$
  $\hat{\mathbf{G}}^t \leftarrow \frac{L_+ - L_-}{2\epsilon\sqrt{r}} \mathbf{Z} \odot \mathbf{M}$ {ZO gradient estimate}
  $\mathbf{W}^{t+1} \leftarrow \mathbf{W}^t - \eta \hat{\mathbf{G}}^t$ {Update the parameters}
end for

supports with the same sparsity level. We therefore study a complementary question: *given the same perturbation density, which coordinates should be selected?* To this end, we characterize sparse support quality through the **Global Energy Capture Ratio**, which measures how much gradient energy is preserved by a given support. We show that, under stated structural and concentration assumptions, WIS can preserve more gradient energy than random masking, and then connect the captured energy to directional alignment and convergence under standard non-convex assumptions.

5.1 Energy Analysis

We first establish the relationship between weight–activation saliency and expected gradient energy under several technical assumptions.

Assumption 5.1 (Weak Correlation)**.** For a linear layer $\mathbf{G}_{i,j} = \boldsymbol{\delta}_i \mathbf{X}_j$, there exist constants $0 < \gamma \leq \nu < \infty$ such that:

$$\gamma \mathbb{E}[\boldsymbol{\delta}_i^2]\mathbb{E}[\mathbf{X}_j^2] \leq \mathbb{E}[\mathbf{G}_{i,j}^2] \leq \nu \mathbb{E}[\boldsymbol{\delta}_i^2]\mathbb{E}[\mathbf{X}_j^2].$$

This assumption characterizes the statistical decoupling between layer inputs $\mathbf{X}$ and backpropagated error signals $\boldsymbol{\delta}$, and is strictly more relaxed than the rigorous independence assumption typically made in previous work (He et al., 2015).

Assumption 5.2 (Normalized Sub-Gaussian Activations)**.** For each input feature $\mathbf{X}_j$, there exists a constant $K \geq 1$ such that

$$\|\mathbf{X}_j\|_{\psi_2}^2 \leq K \mathbf{V}_j, \qquad \mathbf{V}_j = \mathbb{E}[\mathbf{X}_j^2],$$

where $\|\cdot\|_{\psi_2}$ denotes the sub-Gaussian norm. Consequently, for any $\xi \in (0, 1)$, there exists a universal constant $c > 0$ such that

$$\Pr\Big(|\hat{\mathbf{V}}_j - \mathbf{V}_j| \geq \xi \mathbf{V}_j\Big) \leq 2\exp\left(-\frac{cB\xi^2}{K^2}\right).$$

For each row i, let $\mathcal{A}_i$ denote a comparison set containing both the WIS-selected top-k support and the oracle top-k gradient-energy support introduced below. Define $w_{\min} = \min_{j \in \mathcal{A}_i} \mathbf{W}_{i,j}^2 > 0$ and $w_{\max} = \max_{j \in \mathcal{A}_i} \mathbf{W}_{i,j}^2$. The saliency contrast ratios are

$$\kappa \triangleq \frac{\gamma w_{\min}}{\nu w_{\max}}, \qquad \kappa_\xi \triangleq \kappa \frac{1-\xi}{1+\xi}.$$

Here, κ reflects data correlations and weight heterogeneity within the compared supports, while κ_ξ additionally accounts for calibration error.

Lemma 5.3 (Gradient-Saliency Proxy Bound)**.** *Assume Assumptions 5.1 and 5.2 hold. If the calibration batch size satisfies $B = \Omega(K^2 \log n/\xi^2)$, then for any row i and any parameter $j \in \mathcal{A}_i$, the expected gradient energy $\mathbb{E}[\mathbf{G}_{i,j}^2]$ is bounded by the Wanda score $\mathbf{S}_{i,j}^2$ through:*

$$C_{i,\text{low}}\mathbf{S}_{i,j}^2 \le \mathbb{E}[\mathbf{G}_{i,j}^2] \le C_{i,\text{high}}\mathbf{S}_{i,j}^2,$$

where $C_{i,\text{low}}/C_{i,\text{high}} = \kappa_\xi$.

Lemma 5.3 gives a conditional relation between the Wanda score and expected gradient energy within $\mathcal{A}_i$. The factor κ_ξ captures the ranking distortion caused by data correlations, weight heterogeneity, and finite calibration.

Building upon this probabilistic alignment, we formalize the efficiency of WIS-MeZO by introducing the Expected Global Energy Capture Ratio $\rho \in [0, 1]$. This metric quantifies the fraction of gradient information preserved within the selected sub-network:

$$\rho \triangleq \frac{\mathbb{E}[\|\mathbf{M} \odot \mathbf{G}\|^2]}{\mathbb{E}[\|\mathbf{G}\|^2]} = \sum_{i=1}^{m} \omega_i \rho_i, \tag{3}$$

$$\text{where } \omega_i = \frac{\mathbb{E}[\|\mathbf{G}_{i,:}\|^2]}{\mathbb{E}[\|\mathbf{G}\|^2]}, \quad \rho_i = \frac{\mathbb{E}[\|\mathbf{M}_{i,:} \odot \mathbf{G}_{i,:}\|^2]}{\mathbb{E}[\|\mathbf{G}_{i,:}\|^2]},$$

where $\|\cdot\|$ denotes the Frobenius norm, and the expectation is taken over the data distribution. This metric naturally measures how much useful gradient signal is retained after masking, and plays a key role in determining zeroth-order updates. Let ρ^* denote the energy capture ratio of the oracle top-k selection within $\mathcal{A}_i$. Unlike random masking, WIS-MeZO prioritizes components with high saliency scores. The following theorem gives a sufficient condition under which this selection captures more gradient energy than a uniform random mask with the same density r.

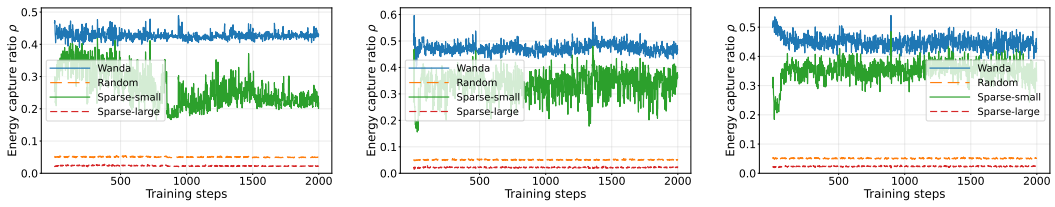
Theorem 5.4 (Energy Concentration Efficiency)**.** *Under the conditions of Lemma 5.3, let r be the density ratio. If $B = \Omega(K^2 \log n/\xi^2)$, then with probability at least $1 - \mathcal{O}(n^{-1})$, the energy capture ratio ρ_{WIS} satisfies:*

$$\rho_{WIS} \ge \kappa_\xi \rho^*.$$

Consequently, if $\rho^* > r/\kappa_\xi$, WIS-MeZO captures more gradient energy than random masking.

The calibration complexity $B = \Omega(K^2 \log n/\xi^2)$ follows from uniform concentration over the layer width. The condition $\rho^* > r/\kappa_\xi$ makes explicit that the advantage depends jointly on gradient-energy concentration and the accuracy of the WIS ranking.

Under this condition, Theorem 5.4 formalizes that WIS allocates sparse perturbations to coordinates carrying more gradient energy than random masking.



(a) RTE (b) QQP (c) MNLI

Figure 2: **Gradient energy capture ratio ρ on LLaMA-3.2-3B at a density ratio of $r = 0.05$.** We compare uniform random masking, the small-magnitude mask used to represent the selection principle of Sparse-MeZO (W-small), the corresponding large-magnitude mask (W-large), and the Wanda-based WIS mask.

Empirical Evaluation of Gradient Energy Preservation. Figure 2 compares four perturbation supports on LLaMA-3.2-3B at $r = 0.05$. Using mini-batch gradients from an auxiliary AdamW diagnostic run with masks fixed at step 0, WIS consistently preserves over 40% of the gradient energy, compared with the 5% expected under random masking. Thus, $\rho_{\text{WIS}} > r$ empirically, consistent with the qualitative implication of Theorem 5.4. WIS also captures more energy than W-small and

W-large, showing that downstream activation statistics provide information beyond weight magnitude alone. W-small outperforms random masking, whereas W-large performs worse, supporting the small-magnitude perturbation principle of Sparse-MeZO (Liu et al., 2025) and showing that the observed gradient energy is not monotonically increasing with weight magnitude in the evaluated settings. Additional density ratios are reported in Appendix C.1.

Before deriving the convergence rate, we analyze the geometric quality of the gradient estimator $\hat{\mathbf{G}}$ produced by WIS-MeZO. In zeroth-order optimization, the efficiency of the update is determined by the alignment between the stochastic estimator $\hat{\mathbf{G}}$ and the true descent direction $\mathbf{G}$. We quantify this alignment using the expected cosine similarity.

Lemma 5.5 (Directional Alignment Efficiency)**.** *Under the conditions of Lemma 5.3, let $\hat{\mathbf{G}}$ be the WIS-MeZO gradient estimator. We define a normalized alignment metric (serving as a proxy for cosine similarity) between $\hat{\mathbf{G}}$ and the stochastic gradient $\mathbf{G}$ by:*

$$\mathrm{Cos}(\hat{\mathbf{G}}, \mathbf{G}) \triangleq \frac{\mathbb{E}[\langle \hat{\mathbf{G}}, \mathbf{G} \rangle]}{\sqrt{\mathbb{E}[\|\hat{\mathbf{G}}\|^2] \cdot \mathbb{E}[\|\mathbf{G}\|^2]}} = \sqrt{\frac{\rho}{mk + \mu_4 - 1}},$$

where $\mu_4 = \mathbb{E}[\mathbf{Z}_{i,j}^4]$ and k is the number of parameters updated per row. A random mask replaces ρ by r, giving a relative alignment factor of $\sqrt{\rho/r}$.

The lemma shows that sparsity alone does not determine estimator quality: at the same sparse dimension, a support with a larger energy capture ratio achieves better expected alignment with the stochastic gradient.

5.2 Convergence Analysis

In this section, we provide a theoretical analysis of the convergence of WIS-MeZO for standard non-convex problems; prior to the analysis, we gather several commonly adopted assumptions.

Assumption 5.6 (L-smoothness)**.** The objective function f is L-smooth, i.e., $\|\nabla f(\mathbf{W}) - \nabla f(\mathbf{W}')\| \leq L\|\mathbf{W} - \mathbf{W}'\|$ for all $\mathbf{W}, \mathbf{W}' \in \mathbb{R}^{m \times n}$.

Assumption 5.7 (Bounded Variance)**.** We assume the stochastic gradient $\mathbf{G}^t := \nabla f(\mathbf{W}^t, \mathcal{B}^t)$ computed on a mini-batch $\mathcal{B}^t$ is an unbiased estimator of the true full-batch gradient $\bar{\mathbf{G}}^t := \nabla f(\mathbf{W}^t)$ at iteration t, satisfying $\mathbb{E}_{\mathcal{B}^t}[\mathbf{G}^t] = \bar{\mathbf{G}}^t$. Furthermore, its variance is bounded by a constant $\sigma^2 \geq 0$:

$$\mathbb{E}_{\mathcal{B}^t}[\|\mathbf{G}^t - \bar{\mathbf{G}}^t\|^2] \leq \sigma^2.$$

Theorem 5.8 (General Convergence Result)**.** *Let $d = mn$, $d_{\text{eff}} = mnr = mk$, and*

$$\widetilde{d} \triangleq \frac{d_{\text{eff}} + \mu_4 - 1}{r}, \qquad \mu_4 = \mathbb{E}[\mathbf{Z}_{i,j}^4].$$

Define

$$\rho_t \triangleq \frac{\|\mathbf{M} \odot \nabla f(\mathbf{W}^t)\|_F^2}{\|\nabla f(\mathbf{W}^t)\|_F^2}, \qquad \rho_{\min} \triangleq \min_{0 \leq t < T} \rho_t > 0.$$

Under Assumptions 5.6 and 5.7, if the learning rate satisfies $\eta_t = 1/(L\widetilde{d}\sqrt{t+1})$, then the iterates $\{\mathbf{W}^t\}_{0 \leq t \leq T-1}$ generated by WIS-MeZO satisfy

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}[\|\nabla f(\mathbf{W}^t)\|^2] = \mathcal{O}\left(\frac{L(d_{\text{eff}} + \mu_4 - 1)}{\rho_{\min}\sqrt{T}}\right) + \mathcal{O}\left(\frac{\sigma^2 \log T}{\rho_{\min}\sqrt{T}}\right).$$

The bound depends on both the effective dimension d_{eff} and the minimum captured gradient energy $\rho_{\min}$. This analysis applies to fixed-mask WIS-MeZO; WISD-MeZO is treated as an empirical extension with time-varying supports.

6 Experiments

In this section, we empirically demonstrate the effectiveness of WIS-MeZO through extensive experiments. To ensure an equitable and fair comparison with related works, our evaluation benchmarks encompass both the GLUE (Wang et al., 2018) and SuperGLUE (Wang et al., 2019) benchmarks. In addition, we assess the memory efficiency and computational throughput of our method.

Table 1: **Experimental results in GLUE benchmark (5 random seeds).** We report F1 scores for QQP, MRPC, and SQuAD, and accuracy (%) for all other tasks (↑). FT and LoRA denote full parameter fine-tuning and Low-Rank Adaptation, respectively, both utilizing the AdamW optimizer.

Models	Methods	MNLI	RTE	QQP	MRPC	SST-2	SQuAD	Avg.
LLaMA3.2-3B	Zero-shot	38.3	49.4	53.7	78.3	58.8	55.0	55.6
	FT	$82.6_{\pm 0.6}$	$87.3_{\pm 0.8}$	$79.2_{\pm 1.0}$	$86.9_{\pm 1.0}$	$95.1_{\pm 0.5}$	$88.2_{\pm 0.4}$	86.5
	LoRA	$83.4_{\pm 0.7}$	$88.8_{\pm 0.7}$	$79.2_{\pm 1.0}$	$85.8_{\pm 1.3}$	$94.8_{\pm 0.1}$	$88.0_{\pm 0.0}$	86.6
	MeZO	$49.4_{\pm 3.0}$	$70.3_{\pm 2.1}$	$65.0_{\pm 2.1}$	$81.4_{\pm 0.6}$	$93.9_{\pm 0.8}$	$86.7_{\pm 0.6}$	74.4
	Sparse-MeZO	$56.7_{\pm 2.6}$	$77.6_{\pm 3.8}$	$70.6_{\pm 0.6}$	$81.8_{\pm 0.9}$	$93.6_{\pm 0.3}$	$86.4_{\pm 0.7}$	77.7
	Random-MeZO	$52.1_{\pm 1.8}$	$74.3_{\pm 0.7}$	$67.4_{\pm 1.3}$	$82.6_{\pm 1.1}$	$93.9_{\pm 0.4}$	$88.1_{\pm 0.6}$	76.4
	WIS-MeZO	$61.2_{\pm 1.1}$	$78.3_{\pm 1.3}$	$68.7_{\pm 1.0}$	$\mathbf{83.1}_{\pm 1.1}$	$\mathbf{94.0}_{\pm 0.2}$	$89.2_{\pm 0.8}$	79.1
	WISD-MeZO	$\mathbf{61.4}_{\pm 1.3}$	$\mathbf{80.1}_{\pm 2.2}$	$\mathbf{71.2}_{\pm 0.9}$	$82.9_{\pm 1.0}$	$\mathbf{94.0}_{\pm 0.5}$	$\mathbf{89.4}_{\pm 1.6}$	**79.8**
Qwen2.5-3B	Zero-shot	74.8	80.5	34.1	51.1	56.0	83.6	63.4
	FT	$86.9_{\pm 0.5}$	$87.8_{\pm 0.6}$	$81.2_{\pm 0.1}$	$89.1_{\pm 0.8}$	$94.1_{\pm 0.4}$	$85.6_{\pm 0.2}$	87.5
	LoRA	$86.6_{\pm 0.4}$	$87.9_{\pm 0.4}$	$79.5_{\pm 0.2}$	$89.6_{\pm 1.0}$	$93.6_{\pm 0.4}$	$87.2_{\pm 1.0}$	87.4
	MeZO	$84.8_{\pm 0.7}$	$84.8_{\pm 0.9}$	$78.7_{\pm 1.1}$	$84.9_{\pm 1.8}$	$92.6_{\pm 0.2}$	$86.6_{\pm 0.9}$	85.4
	Sparse-MeZO	$86.1_{\pm 1.0}$	$86.6_{\pm 1.5}$	$81.3_{\pm 0.8}$	$85.9_{\pm 1.0}$	$93.6_{\pm 0.6}$	$86.3_{\pm 0.7}$	86.6
	Random-MeZO	$85.1_{\pm 1.3}$	$85.2_{\pm 1.6}$	$81.3_{\pm 0.6}$	$84.9_{\pm 1.1}$	$93.1_{\pm 0.2}$	$86.6_{\pm 0.6}$	86.0
	WIS-MeZO	$\mathbf{86.7}_{\pm 0.4}$	$\mathbf{87.0}_{\pm 0.7}$	$81.5_{\pm 0.6}$	$85.6_{\pm 1.1}$	$93.6_{\pm 0.2}$	$\mathbf{87.1}_{\pm 0.8}$	86.9
	WISD-MeZO	$86.3_{\pm 0.9}$	$\mathbf{87.0}_{\pm 1.0}$	$\mathbf{81.6}_{\pm 0.7}$	$\mathbf{86.4}_{\pm 1.2}$	$\mathbf{93.7}_{\pm 0.1}$	$86.8_{\pm 1.1}$	**87.0**

6.1 Fine-tuning on GLUE and SuperGLUE Benchmarks

We show that WIS-MeZO achieves strong aggregate performance compared with the evaluated ZO baselines for downstream LLM fine-tuning.

Setup. We evaluate fixed-mask WIS-MeZO ($\tau = T$ in Algorithm 1) and its dynamic variant WISD-MeZO ($\tau < T$) on LLaMA-3.2-3B (Grattafiori et al., 2024) and Qwen2.5-3B (Yang et al., 2024) using GLUE and SuperGLUE. Hyperparameter tuning and implementation details are provided in the Appendix.

Baselines. We compare against both first-order (FO) and zeroth-order (ZO) methods. FO baselines include full-parameter fine-tuning (**FT**) with AdamW (Kingma & Ba, 2015; Loshchilov & Hutter, 2019) and **LoRA** (Hu et al., 2022). ZO baselines include **MeZO** (Malladi et al., 2023), **Sparse-MeZO** (Liu et al., 2025), which selects sparse perturbations using pretrained weight magnitudes, and **Random-MeZO**, which uses random sparse masks.

Main Results. Tables 1 and 2 report results on GLUE and SuperGLUE, respectively. Figure 1 further compares the validation-loss curves of WIS-MeZO and MeZO over five random seeds. We summarize the main observations below.

WIS-MeZO improves ZO fine-tuning performance and convergence. Among the evaluated fixed-mask ZO methods, WIS-MeZO achieves the strongest aggregate results on GLUE and SuperGLUE and matches or outperforms MeZO on every evaluated task. It reduces the relative error by 18.4%/16.5% on LLaMA-3.2-3B and 10.3%/8.0% on Qwen2.5-3B for GLUE/SuperGLUE, respectively. The consistent gains across both model families suggest that the improvement is not specific to a particular backbone. WIS-MeZO also reaches selected validation-loss levels in nearly $2\times$ fewer steps on RTE and MNLI and over $5\times$ fewer steps on BoolQ, consistent with the benefit of higher gradient energy capture.

WIS-guided perturbations outperform task-agnostic masking. WIS achieves stronger aggregate results than random masking and Sparse-MeZO, indicating that the gain cannot be explained by sparsity alone. Instead, downstream activation statistics provide more informative perturbation supports than random or magnitude-only selection.

Dynamic mask refresh provides modest additional gains. WISD-MeZO further improves several settings, while fixed-mask WIS-MeZO remains the default because the additional gains are modest relative to the extra mask recomputation and tuning of τ.

Table 2: **Experimental results on SuperGLUE.** WIS-based methods achieve the strongest aggregate ZO performance on both model families.

Models	Methods	CB	MultiRC	WSC	ReCoRD	BoolQ	Avg.
LLaMA3.2-3B	Zero-shot	44.6	47.0	36.5	79.7	61.6	53.9
	FT	91.0	83.6	57.6	81.6	83.1	79.4
	LoRA	92.8	85.2	62.5	81.4	84.3	81.2
	MeZO	71.4	72.4	61.5	81.2	77.6	72.8
	Sparse-MeZO	67.8	**79.7**	62.5	81.4	81.3	74.5
	Random-MeZO	75.0	74.2	61.5	81.0	78.9	74.1
	WIS-MeZO	**82.1**	76.3	63.4	**82.0**	82.6	77.3
	WISD-MeZO	**82.1**	76.3	**64.4**	81.9	**83.0**	**77.6**
Qwen2.5-3B	Zero-shot	30.3	65.8	63.4	72.1	70.3	60.3
	FT	96.4	88.0	68.4	74.4	80.9	81.6
	LoRA	94.6	86.9	67.4	76.0	85.1	82.0
	MeZO	96.4	83.2	66.3	73.7	80.9	80.1
	Sparse-MeZO	92.8	86.1	67.3	**75.6**	80.1	80.3
	Random-MeZO	96.4	85.6	65.3	74.5	81.6	80.9
	WIS-MeZO	96.4	86.4	68.2	74.9	82.6	81.7
	WISD-MeZO	**98.2**	**86.9**	**71.1**	**75.6**	**82.8**	**82.9**

6.2 Scaling to Large-Scale LLMs

We further evaluate fixed-mask WIS-MeZO on Qwen2.5 models ranging from 3B to 32B parameters using ARC-E, ARC-C (Clark et al., 2018), HellaSwag (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2021), and OpenBookQA (Mihaylov et al., 2018). As shown in Table 3, WIS-MeZO improves the average score at every scale and matches or outperforms MeZO on every reported task. It also consistently improves over the zero-shot WinoGrande baseline, whereas MeZO does not.

Table 3: **Experimental results for Qwen2.5-3B to 32B.** WIS-MeZO consistently improves average performance over MeZO across all model scales.

Models	Methods	ARC-E	ARC-C	HellaSwag	WinoGrande	OpenBookQA	Avg.
Qwen2.5-3B	Zero-shot	70.0	48.4	68.0	66.3	40.8	58.7
	MeZO	82.9	57.8	69.9	66.3	49.0	65.1
	WIS-MeZO	**85.4**	**60.8**	**74.4**	**69.7**	**54.0**	**68.8**
Qwen2.5-7B	Zero-shot	74.9	52.5	73.8	68.9	43.8	62.7
	MeZO	83.0	59.2	74.9	66.5	51.2	66.9
	WIS-MeZO	**85.0**	**61.2**	**79.4**	**70.7**	**53.8**	**70.0**
Qwen2.5-14B	Zero-shot	75.4	53.8	75.8	70.6	43.2	63.7
	MeZO	84.0	67.2	77.2	73.8	57.2	71.8
	WIS-MeZO	**87.8**	**70.9**	**81.5**	**76.6**	**58.8**	**75.1**
Qwen2.5-32B	Zero-shot	74.5	54.5	79.0	74.3	43.6	65.1
	MeZO	85.9	**69.8**	78.7	74.9	52.4	72.3
	WIS-MeZO	**88.7**	**69.8**	**84.5**	**77.9**	**56.2**	**75.4**

7 Conclusion

In this paper, we present WIS-MeZO, a task-conditioned sparse ZO method that uses weight–activation saliency to allocate perturbations while keeping the pretrained model dense. We characterize sparse support quality through gradient energy capture and establish an L-smooth convergence result linking optimization behavior to both the effective search dimension and captured gradient energy. Our energy analysis further shows that WIS supports generally capture more gradient energy than random and magnitude-only supports, highlighting the benefit of task-dependent activation information for perturbation selection. Across 16 benchmarks and LLaMA and Qwen models ranging from 3B to 32B parameters, WIS-MeZO improves aggregate performance over MeZO and random sparse perturbations, remains competitive with or outperforms Sparse-MeZO, and reaches comparable validation losses in fewer optimization steps. These results demonstrate the effectiveness of activation-aware support selection for backpropagation-free LLM fine-tuning.

AI USE STATEMENT

In this work, we used generative AI tools to edit the manuscript for clarity, conciseness, grammar, and English-language presentation. We have not used generative AI tools for other research tasks requiring disclosure, and the remaining disclosure categories are not applicable to this work. We have reviewed all AI-assisted edits and manually verified the final manuscript. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

REPRODUCIBILITY STATEMENT

We provide the information required to reproduce the proposed method and experiments throughout the paper and appendix. The assumptions and complete derivations supporting the theoretical results are also provided in the appendix. We will make the source code and configuration files publicly available upon acceptance.

REFERENCES

Stephen Bach, Victor Sanh, Zheng-Xin Yong, Albert Webson, Colin Raffel, Nihal V Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Fevry, et al. Promptsource: An integrated development environment and repository for natural language prompts. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pp. 93–104, 2022.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

HanQin Cai, Daniel McKenzie, Wotao Yin, and Zhenliang Zhang. Zeroth-order regularized optimization (zoro): Approximately sparse gradients and adaptive sampling. *SIAM Journal on Optimization*, 32(2):687–714, 2022.

Aochuan Chen, Yimeng Zhang, Jinghan Jia, James Diffenderfer, Jiancheng Liu, Konstantinos Parasyris, Yihua Zhang, Zheng Zhang, Bhavya Kailkhura, and Sijia Liu. Deepzero: Scaling up zeroth-order optimization for deep model training. *arXiv preprint arXiv:2310.02025*, 2023.

Xiangyi Chen, Sijia Liu, Kaidi Xu, Xingguo Li, Xue Lin, Mingyi Hong, and David Cox. Zo-adamm: Zeroth-order adaptive momentum method for black-box optimization. *Advances in neural information processing systems*, 32, 2019.

Yiming Chen, Yuan Zhang, Liyuan Cao, Kun Yuan, and Zaiwen Wen. Enhancing zeroth-order finetuning for language models with low-rank structures. In *The Thirteenth International Conference on Learning Representations*, 2025.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Sizhe Dang, yangyangGuo, Yanjun Zhao, Haishan Ye, xiaodong zheng, Guang Dai, and Ivor Tsang. FZOO: Fast zeroth-order optimizer for fine-tuning large language models towards adam-scale speed. In *The Fourteenth International Conference on Learning Representations*, 2026.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. Gpt3. int8 (): 8-bit matrix multiplication for transformers at scale. *Advances in neural information processing systems*, 35: 30318–30332, 2022.

Utku Evci, Trevor Gale, Jacob Menick, Pablo Samuel Castro, and Erich Elsen. Rigging the lottery: Making all tickets winners. In *International conference on machine learning*, pp. 2943–2952. PMLR, 2020.

Elias Frantar and Dan Alistarh. Sparsegpt: Massive language models can be accurately pruned in one-shot. In *International conference on machine learning*, pp. 10323–10337. PMLR, 2023.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. OPTQ: Accurate quantization for generative pre-trained transformers. In *The Eleventh International Conference on Learning Representations*, 2023.

Tanmay Gautam, Youngsuk Park, Hao Zhou, Parameswaran Raman, and Wooseok Ha. Variance-reduced zeroth-order methods for fine-tuning language models. *arXiv preprint arXiv:2404.08080*, 2024.

Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM journal on optimization*, 23(4):2341–2368, 2013.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Jiaqi Gu, Chenghao Feng, Zheng Zhao, Zhoufeng Ying, Ray T Chen, and David Z Pan. Efficient on-chip learning for optical neural networks through power-aware sparse zeroth-order optimization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 7583–7591, 2021.

Wentao Guo, Jikai Long, Yimeng Zeng, Zirui Liu, Xinyu Yang, Yide Ran, Jacob R. Gardner, Osbert Bastani, Christopher De Sa, Xiaodong Yu, Beidi Chen, and Zhaozhuo Xu. Zeroth-order fine-tuning of LLMs with transferable static sparsity. In *The Thirteenth International Conference on Learning Representations*, 2025.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

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. In *International Conference on Learning Representations*, 2022.

Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International conference on machine learning*, pp. 2137–2146. PMLR, 2018.

Taejong Joo, Wenhan Xia, Cheolmin Kim, Ming Zhang, and Eugene Ie. On surprising effectiveness of masking updates in adaptive optimizers. *arXiv preprint arXiv:2602.15322*, 2026.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Wei Lin, Yining Jiang, Qingyu Song, Qiao Xiang, and Hong Xu. Agzo: Activation-guided zeroth-order optimization for llm fine-tuning. *arXiv preprint arXiv:2601.17261*, 2026.

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. In *Forty-first International Conference on Machine Learning*, 2024.

Yong Liu, Zirui Zhu, Chaoyu Gong, Minhao Cheng, Cho-Jui Hsieh, and Yang You. Sparse meZO: Less parameters for better performance in zeroth-order LLM fine-tuning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Sadhika Malladi, Tianyu Gao, Eshaan Nichani, Alex Damian, Jason D Lee, Danqi Chen, and Sanjeev Arora. Fine-tuning language models with just forward passes. *Advances in Neural Information Processing Systems*, 36:53038–53075, 2023.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 2381–2391, 2018.

Mayumi Ohta, Nathaniel Berger, Artem Sokolov, and Stefan Riezler. Sparse perturbations for improved convergence in stochastic zeroth-order optimization. In *International Conference on Machine Learning, Optimization, and Data Science*, pp. 39–64. Springer, 2020.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

James C Spall. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE transactions on automatic control*, 37(3):332–341, 2002.

Mingjie Sun, Zhuang Liu, Anna Bair, and J Zico Kolter. A simple and effective pruning approach for large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP workshop BlackboxNLP: Analyzing and interpreting neural networks for NLP*, pp. 353–355, 2018.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32, 2019.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. Smoothquant: Accurate and efficient post-training quantization for large language models. In *International conference on machine learning*, pp. 38087–38099. PMLR, 2023.

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, 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. *arXiv preprint arXiv:2412.15115*, 2024.

Ziming Yu, Pan Zhou, Sike Wang, Jia Li, Mi Tian, and Hua Huang. Zeroth-order fine-tuning of llms in random subspaces. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4475–4485, 2025.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 4791–4800, 2019.

Yihua Zhang, Pingzhi Li, Junyuan Hong, Jiaxiang Li, Yimeng Zhang, Wenqing Zheng, Pin-Yu Chen, Jason D. Lee, Wotao Yin, Mingyi Hong, Zhangyang Wang, Sijia Liu, and Tianlong Chen. Revisiting zeroth-order optimization for memory-efficient llm fine-tuning: a benchmark. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Zhen Zhang, Yifan Yang, Kai Zhen, Nathan Susanj, Athanasios Mouchtaris, Siegfried Kunzmann, and Zheng Zhang. Mazo: Masked zeroth-order optimization for multi-task fine-tuning of large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 18537–18554, 2025.

Hanzhen Zhao, Shihong Ding, Cong Fang, and Zhouchen Lin. Pazo: Preconditioned accelerated zeroth-order optimization for fine-tuning llms. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025a.

Jiawei Zhao, Zhenyu Zhang, Beidi Chen, Zhangyang Wang, Anima Anandkumar, and Yuandong Tian. Galore: Memory-efficient llm training by gradient low-rank projection, 2024.

Yanjun Zhao, Sizhe Dang, Haishan Ye, Guang Dai, Yi Qian, and Ivor Tsang. Second-order fine-tuning without pain for LLMs: A hessian informed zeroth-order optimizer. In *The Thirteenth International Conference on Learning Representations*, 2025b.

Hanqing Zhu, Zhenyu Zhang, Wenyan Cong, Xi Liu, Sem Park, Vikas Chandra, Bo Long, David Z. Pan, Zhangyang Wang, and Jinwon Lee. APOLLO: SGD-like memory, adamw-level performance. In *Eighth Conference on Machine Learning and Systems*, 2025.

APPENDIX CONTENTS

Appendix Contents **14**

A Proofs **15**

A.1 List of Notations 15

A.2 Wanda Score as a Gradient Proxy 15

A.3 Proof of Lemma 5.3 16

A.4 Proof of Theorem 5.4 16

A.5 Proof of Lemma 5.5 17

A.6 Supplementary PL Result 18

A.7 Proof of Theorem A.2 18

A.8 Proof of Theorem 5.8 18

B Experimental Details **19**

C Additional Experimental Results **19**

C.1 Additional Gradient Energy Capture Results 19

C.2 Memory and Speed Comparison 23

C.3 Study on Density Ratio-r 24

C.4 Effects of Calibration Size and Computation Cost 24

D Detailed Pseudocode **25**

D.1 Layer-Wise Parameter Perturbation 25

D.2 WIS Mask Construction 25

E Prompts **26**

A PROOFS

A.1 LIST OF NOTATIONS

Table 4 summarizes the key symbols and notations used throughout this paper to facilitate the understanding of our theoretical framework.

Table 4: Summary of key notations and symbols.

Symbol	Description
$\mathbf{W}, \mathbf{Z}, \mathbf{G} \in \mathbb{R}^{m\times n}$	Weight, perturbation, and stochastic-gradient matrices.
$\hat{\mathbf{G}} \in \mathbb{R}^{m\times n}$	Zeroth-order gradient estimator.
$\mathbf{X} \in \mathbb{R}^{n\times B}$	Input activations from B calibration samples.
$\mathbf{S}_{i,j}$	Wanda importance score for $\mathbf{W}_{i,j}$.
$\mathbf{V}_j, \hat{\mathbf{V}}_j$	Population and empirical second moments of feature j.
$\mathbf{M} \in \{0,1\}^{m\times n}$	Row-wise top-k perturbation mask.
$\mathcal{A}_i$	Comparison set containing the WIS and oracle top-k supports for row i.
$r,\ k = rn$	Density ratio and selected coordinates per row.
$d = mn,\ d_{\text{eff}} = mk$	Ambient and effective perturbation dimensions.
γ, ν	Weak-correlation constants in Assumption 5.1.
K	Normalized sub-Gaussian constant in Assumption 5.2.
ξ, κ, κ_ξ	Calibration error and saliency contrast ratios.
$\rho, \rho^*, \rho_{\min}$	Captured, oracle, and minimum captured energy ratios.
$C_{i,\text{low}}, C_{i,\text{high}}$	Gradient–saliency proxy constants.
ϵ, η, τ	Perturbation scale, learning rate, and mask refresh interval.
$\mu_4 = \mathbb{E}[\mathbf{Z}_{i,j}^4]$	Fourth moment of the perturbation distribution.
L, μ, σ^2	Smoothness, PL, and stochastic-variance constants.
$\|\cdot\|_2, \|\cdot\|_F$	Euclidean and Frobenius norms.
$\odot$	Element-wise multiplication.

A.2 WANDA SCORE AS A GRADIENT PROXY

Consider a linear layer evaluated on a randomly drawn downstream example,

$$\mathbf{y} = \mathbf{W}\mathbf{x}, \qquad \mathbf{W} \in \mathbb{R}^{d_{\text{out}}\times d_{\text{in}}}, \qquad \mathbf{x} \in \mathbb{R}^{d_{\text{in}}}.$$

Let $\boldsymbol{\delta}_i := \partial f/\partial \mathbf{y}_i$ denote the backpropagated signal. The per-example gradient contribution for $\mathbf{W}_{i,j}$ is

$$\mathbf{G}_{i,j} := \frac{\partial f}{\partial \mathbf{W}_{i,j}} = \boldsymbol{\delta}_i \mathbf{x}_j.$$

The analysis below concerns the population energy $\mathbb{E}[\mathbf{G}_{i,j}^2]$ over the downstream data distribution. We use the per-example contribution in the proxy analysis because the square of an aggregated mini-batch gradient contains cross-example terms that require additional assumptions. The empirical study in Appendix C.1 instead measures energy capture directly on the actual mini-batch gradients used in an auxiliary AdamW run, thereby testing whether the support alignment persists after gradient aggregation.

To construct the mask, we collect B calibration activations and write $\mathbf{X} = [\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(B)}] \in \mathbb{R}^{d_{\text{in}}\times B}$. The empirical and population second moments of feature j are

$$\hat{\mathbf{V}}_j := \frac{1}{B}\sum_{b=1}^{B} (\mathbf{X}_j^{(b)})^2, \qquad \mathbf{V}_j := \mathbb{E}[\mathbf{x}_j^2].$$

The Wanda score is

$$\mathbf{S}_{i,j} := |\mathbf{W}_{i,j}| \, \|\mathbf{X}_j\|_2, \qquad \|\mathbf{X}_j\|_2 = \sqrt{B\hat{\mathbf{V}}_j},$$

and therefore

$$\mathbf{S}_{i,j}^2 = B\mathbf{W}_{i,j}^2 \hat{\mathbf{V}}_j.$$

A.3 Proof of Lemma 5.3

Proof. For a randomly drawn downstream example, $\mathbf{G}_{i,j} = \boldsymbol{\delta}_i \mathbf{X}_j$. Under Assumption 5.1,

$$\gamma \, \mathbb{E}[\boldsymbol{\delta}_i^2]\mathbf{V}_j \le \mathbb{E}[\mathbf{G}_{i,j}^2] \le \nu \, \mathbb{E}[\boldsymbol{\delta}_i^2]\mathbf{V}_j. \tag{4}$$

Assumption 5.2 and a union bound over the n input features imply that, when $B = \Omega(K^2 \log n/\xi^2)$, the event

$$(1-\xi)\mathbf{V}_j \le \hat{\mathbf{V}}_j \le (1+\xi)\mathbf{V}_j, \qquad j = 1, \dots, n,$$

holds with probability at least $1 - \mathcal{O}(n^{-1})$. Since $\mathbf{S}_{i,j}^2 = B\mathbf{W}_{i,j}^2\hat{\mathbf{V}}_j$, this event gives

$$\frac{\mathbf{S}_{i,j}^2}{B(1+\xi)\mathbf{W}_{i,j}^2} \le \mathbf{V}_j \le \frac{\mathbf{S}_{i,j}^2}{B(1-\xi)\mathbf{W}_{i,j}^2}. \tag{5}$$

For $j \in \mathcal{A}_i$, we have $w_{\min} \le \mathbf{W}_{i,j}^2 \le w_{\max}$. Combining the upper bounds in Eqs. equation 4 and equation 5 yields

$$\mathbb{E}[\mathbf{G}_{i,j}^2] \le \frac{\nu \mathbb{E}[\boldsymbol{\delta}_i^2]}{B(1-\xi)w_{\min}} \mathbf{S}_{i,j}^2 \triangleq C_{i,\text{high}} \mathbf{S}_{i,j}^2.$$

Similarly,

$$\mathbb{E}[\mathbf{G}_{i,j}^2] \ge \frac{\gamma \mathbb{E}[\boldsymbol{\delta}_i^2]}{B(1+\xi)w_{\max}} \mathbf{S}_{i,j}^2 \triangleq C_{i,\text{low}} \mathbf{S}_{i,j}^2.$$

Therefore,

$$\frac{C_{i,\text{low}}}{C_{i,\text{high}}} = \frac{\gamma w_{\min}}{\nu w_{\max}} \frac{1-\xi}{1+\xi} = \kappa_\xi.$$

This proves the result. □

A.4 Proof of Theorem 5.4

Proof. Let $\mathcal{I}_{\text{WIS},i}$ be the set of $k = r \cdot n$ indices selected by WIS-MeZO for row i, and let $\mathcal{I}_i^* \subseteq \mathcal{A}_i$ denote the oracle set containing the k largest expected gradient energies within the active set. By the definition of the oracle selection, the captured energy ratio ρ_i^* is maximal

$$\rho_i^* = \frac{\sum_{j \in \mathcal{I}_i^*} \mathbb{E}[\mathbf{G}_{i,j}^2]}{\mathbb{E}[\|\mathbf{G}_{i,:}\|^2]} \ge \frac{\sum_{j \in \mathcal{I}} \mathbb{E}[\mathbf{G}_{i,j}^2]}{\mathbb{E}[\|\mathbf{G}_{i,:}\|^2]}, \quad \forall \mathcal{I} \subseteq \mathcal{A}_i, \ |\mathcal{I}| = k.$$

For any $j_1 \in \mathcal{I}_{\text{WIS},i}$ and $j_2 \notin \mathcal{I}_{\text{WIS},i}$ where $j_1, j_2 \in \mathcal{A}_i$, the selection criterion ensures $\mathbf{S}_{i,j_1}^2 \ge \mathbf{S}_{i,j_2}^2$. Applying the proxy bounds from Lemma 5.3, we relate the gradient energies of these indices:

$$\mathbb{E}[\mathbf{G}_{i,j_1}^2] \ge C_{i,\text{low}} \mathbf{S}_{i,j_1}^2 \ge C_{i,\text{low}} \mathbf{S}_{i,j_2}^2 \ge \frac{C_{i,\text{low}}}{C_{i,\text{high}}} \mathbb{E}[\mathbf{G}_{i,j_2}^2].$$

From the concentration established in Assumption 5.2, for $B = \Omega(K^2 \log n/\xi^2)$, the proxy bounds hold simultaneously with probability at least $1 - \mathcal{O}(n^{-1})$. Thus, within $\mathcal{A}_i$, the energy of a selected component is bounded below by a factor κ_ξ relative to an excluded component.

Because both the WIS-selected and oracle supports are contained in $\mathcal{A}_i$ by construction, summing over the selected indices gives:

$$\rho_{\text{WIS},i} = \frac{\sum_{j \in \mathcal{I}_{\text{WIS},i}} \mathbb{E}[\mathbf{G}_{i,j}^2]}{\mathbb{E}[\|\mathbf{G}_{i,:}\|^2]} \ge \kappa_\xi \cdot \frac{\sum_{j \in \mathcal{I}_i^*} \mathbb{E}[\mathbf{G}_{i,j}^2]}{\mathbb{E}[\|\mathbf{G}_{i,:}\|^2]} = \kappa_\xi \rho_i^*.$$

The global energy capture ratio ρ_{WIS} is the weighted sum across all rows

$$\rho_{\text{WIS}} = \sum_{i=1}^{m} \omega_i \rho_{\text{WIS},i} \geq \kappa_\xi \sum_{i=1}^{m} \omega_i \rho_i^* = \kappa_\xi \rho^*.$$

Finally, for a random mask, the expected capture ratio is identically $\rho_{\text{rand}} = r$. Under the condition $\rho^* > r/\kappa_\xi$, it follows directly that:

$$\rho_{\text{WIS}} \geq \kappa_\xi \rho^* > r.$$

This establishes the stated sufficient condition for a strict improvement over uniform random masking.

The proof is completed.

□

A.5 Proof of Lemma 5.5

Proof. We begin by evaluating the numerator of the similarity metric. By the law of total expectation, we first take the expectation over the random perturbation $\mathbf{Z}$, followed by the mini-batch $\mathcal{B}$.

Recalling the definition $\hat{\mathbf{G}} = \frac{1}{r}\langle \mathbf{G}, \mathbf{Z} \odot \mathbf{M} \rangle (\mathbf{Z} \odot \mathbf{M})$, the expected inner product satisfies:

$$\begin{aligned} \mathbb{E}[\langle \hat{\mathbf{G}}, \mathbf{G} \rangle] &= \mathbb{E}_{\mathcal{B}} \left[\left\langle \mathbb{E}_{\mathbf{Z}} \left[\frac{1}{r} \langle \mathbf{G}, \mathbf{Z} \odot \mathbf{M} \rangle (\mathbf{Z} \odot \mathbf{M}) \right], \mathbf{G} \right\rangle \right] \\ &= \mathbb{E}_{\mathcal{B}} \left[\left\langle \frac{1}{r} (\mathbf{M} \odot \mathbf{G}), \mathbf{G} \right\rangle \right] = \frac{1}{r} \mathbb{E}_{\mathcal{B}} [\|\mathbf{M} \odot \mathbf{G}\|^2]. \end{aligned} \tag{6}$$

Substituting the energy capture ratio $\rho = \mathbb{E}[\|\mathbf{M} \odot \mathbf{G}\|^2]/\mathbb{E}[\|\mathbf{G}\|^2]$, the signal term simplifies to $\frac{\rho}{r}\mathbb{E}[\|\mathbf{G}\|^2]$.

Next, we characterize the second moment of the estimator $\hat{\mathbf{G}}$. For a random vector $\mathbf{Z}$ with i.i.d. components satisfying $\mathbb{E}[\mathbf{Z}_i] = 0, \mathbb{E}[\mathbf{Z}_i^2] = 1$, and $\mathbb{E}[\mathbf{Z}_i^4] = \mu_4$, the second moment of a zero-order estimator restricted to a subspace of dimension d_{sub} is given by

$$\mathbb{E}_{\mathbf{Z}}[\|\hat{\mathbf{G}}\|^2] = \frac{1}{r^2}(d_{\text{sub}} + \mu_4 - 1)\|\mathbf{M} \odot \mathbf{G}\|^2.$$

With $d_{\text{sub}} = mk$, taking expectation over the mini-batch gives

$$\mathbb{E}[\|\hat{\mathbf{G}}\|^2] = \frac{mk + \mu_4 - 1}{r^2} \rho\, \mathbb{E}[\|\mathbf{G}\|^2]. \tag{7}$$

Combining Eqs. equation 6 and equation 7 yields

$$\text{Cos}(\hat{\mathbf{G}}, \mathbf{G}) = \sqrt{\frac{\rho}{mk + \mu_4 - 1}}.$$

To compare WIS-MeZO with random masking, we note that a random row-wise mask provides no information about the gradient magnitude, resulting in an expected energy capture ratio $\rho_{\text{rand}} = \mathbb{E}[\sum_{j \in \mathcal{I}_{\text{rand}}} \mathbf{G}_{i,j}^2]/\mathbb{E}[\|\mathbf{G}\|^2] = r$. Substituting $\rho = r$ into the derived similarity formula gives

$$\text{Cos}(\hat{\mathbf{G}}_{\text{rand}}, \mathbf{G}) = \sqrt{\frac{r}{mk + \mu_4 - 1}}$$

Since $\rho > r$ holds by Theorem 5.4, it follows that $\text{Cos}(\hat{\mathbf{G}}, \mathbf{G}) > \text{Cos}(\hat{\mathbf{G}}_{\text{rand}}, \mathbf{G})$. The directional gain is precisely $\sqrt{\rho/r}$, confirming that importance-based masking significantly mitigates the noise inherent in zero-order gradient estimation.

This completes the proof. □

A.6 Supplementary PL Result

For completeness, we also provide a stronger result under the PL condition.

Assumption A.1 (μ-PL Condition)**.** There exists $\mu > 0$ such that

$$\|\nabla f(\mathbf{W})\|^2 \geq 2\mu(f(\mathbf{W}) - f^*).$$

Theorem A.2 (Convergence under the PL Condition)**.** *Suppose Assumptions 5.6, 5.7, and A.1 hold. Let* $d_{\text{eff}} = mk$, $\widetilde{d} = (d_{\text{eff}} + \mu_4 - 1)/r$, *and* $\eta = \alpha/(L\widetilde{d})$ *with* $\alpha \in (0, 1]$. *Then*

$$\mathbb{E}[f(\mathbf{W}^T) - f^*] \leq \left(1 - \frac{\alpha\mu\rho_{\min}}{L(d_{\text{eff}} + \mu_4 - 1)}\right)^T (f(\mathbf{W}^0) - f^*) + \frac{\alpha\sigma^2}{2\mu\rho_{\min}}.$$

A.7 Proof of Theorem A.2

Proof. Using the same first- and second-moment bounds as in the proof of Theorem 5.8, together with $\eta = \alpha/(L\widetilde{d})$ and $\alpha \leq 1$, gives

$$\mathbb{E}[f(\mathbf{W}^{t+1})] \leq \mathbb{E}[f(\mathbf{W}^t)] - \frac{\alpha\rho_t}{2L(d_{\text{eff}} + \mu_4 - 1)}\mathbb{E}[\|\nabla f(\mathbf{W}^t)\|^2] + \frac{\alpha^2\sigma^2}{2L(d_{\text{eff}} + \mu_4 - 1)}. \tag{8}$$

Applying the PL inequality and defining $\Delta_t := \mathbb{E}[f(\mathbf{W}^t) - f^*]$ yield

$$\Delta_{t+1} \leq \left(1 - \frac{\alpha\mu\rho_t}{L(d_{\text{eff}} + \mu_4 - 1)}\right)\Delta_t + \frac{\alpha^2\sigma^2}{2L(d_{\text{eff}} + \mu_4 - 1)}.$$

Using $\rho_t \geq \rho_{\min}$ and unrolling the recursion gives

$$\begin{aligned}\Delta_T &\leq \left(1 - \frac{\alpha\mu\rho_{\min}}{L(d_{\text{eff}} + \mu_4 - 1)}\right)^T \Delta_0 + \frac{\alpha^2\sigma^2}{2L(d_{\text{eff}} + \mu_4 - 1)} \sum_{i=0}^{T-1} \left(1 - \frac{\alpha\mu\rho_{\min}}{L(d_{\text{eff}} + \mu_4 - 1)}\right)^i \\ &\leq \left(1 - \frac{\alpha\mu\rho_{\min}}{L(d_{\text{eff}} + \mu_4 - 1)}\right)^T \Delta_0 + \frac{\alpha\sigma^2}{2\mu\rho_{\min}}.\end{aligned}$$

This proves the result. □

A.8 Proof of Theorem 5.8

Proof. Let $\mathbf{W}^{t+1} = \mathbf{W}^t - \eta_t\hat{\mathbf{G}}^t$. By L-smoothness,

$$f(\mathbf{W}^{t+1}) \leq f(\mathbf{W}^t) - \eta_t\langle\nabla f(\mathbf{W}^t), \hat{\mathbf{G}}^t\rangle + \frac{L\eta_t^2}{2}\|\hat{\mathbf{G}}^t\|^2. \tag{9}$$

For the masked ZO estimator, the first moment satisfies

$$\mathbb{E}\Big[\langle\nabla f(\mathbf{W}^t), \hat{\mathbf{G}}^t\rangle\Big] = \frac{\rho_t}{r}\|\nabla f(\mathbf{W}^t)\|^2. \tag{10}$$

Using the fourth-moment identity from Lemma 5.5 and Assumption 5.7,

$$\begin{aligned}\mathbb{E}[\|\hat{\mathbf{G}}^t\|^2] &\leq \frac{d_{\text{eff}} + \mu_4 - 1}{r^2}\left(\rho_t\|\nabla f(\mathbf{W}^t)\|^2 + \sigma^2\right) \\ &= \frac{\widetilde{d}}{r}\left(\rho_t\|\nabla f(\mathbf{W}^t)\|^2 + \sigma^2\right).\end{aligned} \tag{11}$$

Taking expectations in Eq. equation 9 and substituting Eqs. equation 10 and equation 11 give

$$\mathbb{E}[f(\mathbf{W}^{t+1})] \leq \mathbb{E}[f(\mathbf{W}^t)] - \frac{\rho_t}{r}\left(\eta_t - \frac{L\widetilde{d}}{2}\eta_t^2\right)\mathbb{E}[\|\nabla f(\mathbf{W}^t)\|^2] + \frac{L\widetilde{d}\eta_t^2}{2r}\sigma^2.$$

For $\eta_t = 1/(L\widetilde{d}\sqrt{t+1})$,

$$\eta_t - \frac{L\widetilde{d}}{2}\eta_t^2 \geq \frac{1}{2L\widetilde{d}\sqrt{t+1}}.$$

Since $r\widetilde{d} = d_{\text{eff}} + \mu_4 - 1$, we obtain

$$\mathbb{E}[f(\mathbf{W}^{t+1})] \le \mathbb{E}[f(\mathbf{W}^t)] - \frac{\rho_t}{2L(d_{\text{eff}} + \mu_4 - 1)\sqrt{t+1}}\mathbb{E}[\|\nabla f(\mathbf{W}^t)\|^2] + \frac{\sigma^2}{2L(d_{\text{eff}} + \mu_4 - 1)(t+1)}.$$

Rearranging, summing over $t = 0, \dots, T-1$, and using $\rho_t \ge \rho_{\min}$ yield

$$\sum_{t=0}^{T-1} \frac{1}{\sqrt{t+1}}\mathbb{E}[\|\nabla f(\mathbf{W}^t)\|^2] \le \frac{2L(d_{\text{eff}} + \mu_4 - 1)\Delta f}{\rho_{\min}} + \frac{\sigma^2(1 + \log T)}{\rho_{\min}},$$

where $\Delta f = f(\mathbf{W}^0) - f^*$. Finally, because $1/\sqrt{t+1} \ge 1/\sqrt{T}$ for every $t < T$,

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}[\|\nabla f(\mathbf{W}^t)\|^2] \le \mathcal{O}\left(\frac{L(d_{\text{eff}} + \mu_4 - 1)}{\rho_{\min}\sqrt{T}}\right) + \mathcal{O}\left(\frac{\sigma^2 \log T}{\rho_{\min}\sqrt{T}}\right).$$

This proves the result. □

B Experimental Details

This section details the experimental configurations, including the computational environment and the comprehensive hyperparameter settings.

We first outline the experimental setup for fine-tuning on the GLUE and SuperGLUE benchmarks. To ensure a fair comparison with existing literature, we adhere strictly to the protocols established in prior works. Specifically, for first-order SFT methods, such as full fine-tuning (FT) and LoRA, we employ a batch size of 8. For ZO methods, we increase the batch size to 16 and train for 20k steps to ensure memory parity across different optimization regimes. The datasets are partitioned by sampling 500 instances for the validation set and 1,000 instances for evaluation. Final results are reported using the checkpoint that achieved peak performance on the validation set. All configurations utilize a default sequence length of 2048. Additionally, we employ a constant learning rate schedule with weight decay set to 0, consistent with standard practices in ZO optimization research. These parameters are precisely aligned across all baselines (Malladi et al., 2023; Liu et al., 2025) to facilitate a direct and equitable comparison.

Regarding hyperparameters, optimization methods often exhibit varying levels of sensitivity. To ensure a robust evaluation, we employ a grid search strategy, the specific search space for which is detailed in Table 5. The final configurations utilized for our primary experiments are summarized in Tables 6 and 7. GLUE results are averaged over five independent random seeds (0–4), whereas SuperGLUE results are reported from a single run. WIS-MeZO uses 128 task-specific calibration examples by default. The examples are obtained from a separate iterator over the corresponding training split and processed in 16 mini-batches of size 8. Although these batches contain labels because the model is executed through the standard teacher-forcing loss interface, mask construction uses only forward-pass input activations under inference mode and does not access loss values, gradients, or label contents. Since calibration and optimization draw from the same training split, calibration examples may reappear in later optimization mini-batches. Calibration is performed at step 0, and the resulting mask remains fixed for WIS-MeZO.

Regarding the computing environment, we use Python 3.10.14 and PyTorch 2.8.0 (Paszke et al., 2019). All experiments were conducted on NVIDIA H20 GPUs.

C Additional Experimental Results

C.1 Additional Gradient Energy Capture Results

This section provides the complete experimental settings and additional results for the gradient energy capture analysis presented in the main text. The analysis is conducted as a separate first-order AdamW diagnostic and does not alter WIS-MeZO or MeZO training. We evaluate four mask construction strategies on LLaMA-3.2-3B:

Table 5: The hyperparameter grids used for the experiments in the main text.

Methods	Hyperparameters	Values
FT	Batch size	8
	Learning rate	$\{1.0e-6, 5.0e-6, 1.0e-5, 5.0e-5\}$
	(β_1, β_2)	(0.9, 0.999)
LoRA	Batch size	8
	Learning rate	$\{1.0e-5, 5.0e-5, 1.0e-4, 5.0e-4\}$
	Rank	8
	Alpha-α	16
	(β_1, β_2)	(0.9, 0.999)
MeZO	Batch size	16
	Learning rate	$\{5.0e-8, 1.0e-7, 2.5e-7, 5.0e-7\}$
	ϵ	$\{1.0e-3, 1.0e-4\}$
Sparse-MeZO	Batch size	16
	Learning rate	$\{5.0e-7, 1.0e-6, 2.5e-6, 5.0e-6\}$
	ϵ	$\{1.0e-3, 1.0e-4\}$
	Density ratio-r	$\{0.05, 0.1, 0.2, 0.4\}$
Random-MeZO	Batch size	16
	Learning rate	$\{5.0e-8, 1.0e-7, 2.5e-7, 5.0e-7\}$
	ϵ	$\{1.0e-3, 1.0e-4\}$
	Density ratio-r	$\{0.05, 0.1, 0.2, 0.4\}$
WIS-MeZO	Batch size	16
	Learning rate	$\{5.0e-8, 1.0e-7, 2.5e-7, 5.0e-7\}$
	ϵ	$\{1.0e-3, 1.0e-4\}$
	Density ratio-r	$\{0.05, 0.1, 0.2, 0.4\}$
WISD-MeZO	Batch size	16
	Learning rate	$\{5.0e-8, 1.0e-7, 2.5e-7, 5.0e-7\}$
	ϵ	$\{1.0e-3, 1.0e-4\}$
	Density ratio-r	$\{0.05, 0.1, 0.2, 0.4\}$
	Frequency-τ	$\{2000, 5000, 10000\}$

Table 6: Used hyperparameters for GLUE experiments.

Models	Methods	**MNLI**	**RTE**	**QQP**	**MRPC**	**SST-2**	**SQuAD**
LLaMA3.2-3B	FT	1.0e-5	1.0e-5	1.0e-5	1.0e-5	1.0e-5	1.0e-5
	LoRA	5.0e-4	5.0e-4	5.0e-4	5.0e-4	1.0e-4	5.0e-5
	MeZO	5.0e-7	2.5e-7	5.0e-7	2.5e-7	5.0e-7	2.5e-7
	Sparse-MeZO	2.5e-6, 0.2	1.0e-6, 0.2	1.0e-6, 0.4	1.0e-6, 0.4	1.0e-6, 0.4	1.0e-6, 0.2
	Random-MeZO	5.0e-7, 0.2	5.0e-7, 0.2	5.0e-7, 0.4	5.0e-7, 0.2	5.0e-7, 0.2	5.0e-7, 0.2
	WIS-MeZO	2.5e-7, 0.2	2.5e-7, 0.2	5.0e-7, 0.4	2.5e-7, 0.1	1.0e-7, 0.1	1.0e-7, 0.1
	WISD-MeZO	2.5e-7, 0.2, 5000	2.5e-7, 0.1, 2000	2.5e-7, 0.2, 10000	2.5e-7, 0.1, 2000	1.0e-7, 0.1, 5000	1.0e-7, 0.1, 5000
Qwen2.5-3B	FT	1.0e-5	1.0e-5	1.0e-5	1.0e-5	1.0e-5	1.0e-5
	LoRA	1.0e-4	1.0e-4	5.0e-5	1.0e-4	1.0e-5	5.0e-4
	MeZO	1.0e-7	1.0e-7	2.5e-7	5.0e-7	5.0e-7	5.0e-7
	Sparse-MeZO	2.5e-6, 0.1	2.5e-6, 0.4	5.0e-6, 0.1	5.0e-6, 0.2	5.0e-6, 0.4	5.0e-6, 0.1
	Random-MeZO	1.0e-7, 0.1	5.0e-7, 0.4	1.0e-7, 0.1	2.5e-7, 0.2	5.0e-7, 0.4	5.0e-7, 0.4
	WIS-MeZO	1.0e-7, 0.2	1.0e-7, 0.1	5.0e-8, 0.4	5.0e-8, 0.2	2.5e-7, 0.4	5.0e-8, 0.05
	WISD-MeZO	5.0e-8, 0.2, 5000	5.0e-8, 0.05, 10000	5.0e-8, 0.4, 2000	5.0e-8, 0.2, 5000	2.5e-7, 0.2, 2000	5.0e-8, 0.05, 5000

- **Random:** uniformly sample $k = \lfloor rn \rfloor$ coordinates from each row;
- **W-small:** select the k coordinates with the smallest weight magnitudes $|\mathbf{W}_{i,j}|$ in each row, following the small-magnitude selection principle of Sparse-MeZO;
- **W-large:** select the k coordinates with the largest weight magnitudes $|\mathbf{W}_{i,j}|$ in each row;
- **WIS:** select the k coordinates with the largest Wanda scores $|\mathbf{W}_{i,j}| \|\mathbf{X}_{j,:}\|_2$ in each row.

Table 7: Used hyperparameters for Super-GLUE experiments.

Models	Methods	**CB**	**MultiRC**	**WSC**	**ReCoRD**	**BoolQ**
LLaMA3.2-3B	FT	1.0e-5	1.0e-5	1.0e-5	1.0e-6	1.0e-5
	LoRA	5.0e-4	1.0e-4	5.0e-4	1.0e-5	5.0e-4
	MeZO	2.5e-7	5.0e-7	2.5e-7	2.5e-7	1.0e-7
	Sparse-MeZO	1.0e-6, 0.4	1.0e-6, 0.2	5.0e-6, 0.4	1.0e-6, 0.2	2.5e-6, 0.2
	Random-MeZO	5.0e-7, 0.2	5.0e-7, 0.2	2.5e-7, 0.1	5.0e-7, 0.4	5.0e-7, 0.2
	WIS-MeZO	2.5e-7, 0.1	2.5e-7, 0.1	5.0e-7, 0.05	1.0e-7, 0.2	1.0e-7, 0.05
	WISD-MeZO	2.5e-7, 0.1, 2000	2.5e-7, 0.1,5000	1.0e-7, 0.05,2000	1.0e-7, 0.2, 10000	1.0e-7, 0.1, 10000
Qwen2.5-3B	FT	5.0e-5	1.0e-5	1.0e-5	1.0e-6	1.0e-5
	LoRA	5.0e-5	5.0e-5	1.0e-5	1.0e-5	1.0e-4
	MeZO	5.0e-7	2.5e-7	2.5e-7	5.0e-7	2.5e-7
	Sparse-MeZO	2.5e-6, 0.2	1.0e-6, 0.4	2.5e-6, 0.2	5.0e-6, 0.2	2.5e-6, 0.2
	Random-MeZO	1.0e-7, 0.1	5.0e-7, 0.4	2.5e-7, 0.2	5.0e-7, 0.2	1.0e-7, 0.1
	WIS-MeZO	1.0e-7, 0.1	5.0e-8, 0.05	1.0e-7, 0.2	1.0e-7, 0.1	1.0e-7, 0.2
	WISD-MeZO	2.5e-7, 0.2, 2000	2.5e-7, 0.4, 5000	1.0e-7, 0.2, 10000	1.0e-7, 0.1, 2000	1.0e-7, 0.2,10000

Gradient snapshots. For each task, we run AdamW for 2,000 optimization steps and record one gradient snapshot at every step, yielding 2,000 snapshots in total. Each snapshot is read from `p.grad` after backpropagation and before `optimizer.step()`. We use a per-device batch size of 8 and no gradient accumulation, so $\mathbf{G}^t$ is the mini-batch average gradient from eight training examples. For a fixed mask $\mathbf{M}$, the step-wise capture ratio is

$$\rho_t(\mathbf{M}) = \frac{\|\mathbf{M} \odot \mathbf{G}^t\|_F^2}{\|\mathbf{G}^t\|_F^2}. \tag{12}$$

The reported mean is the arithmetic average over the 2,000 steps; moving averages are used only to visualize the trajectories. The masks are constructed once at step 0 and remain fixed throughout the diagnostic run.

Post-hoc diagnostic. The full gradients used in this experiment are obtained only through the auxiliary AdamW run. They are never used to construct the WIS mask or to update WIS-MeZO or MeZO. Thus, this analysis is a post-hoc diagnostic of whether a fixed sparse support captures the gradient energy encountered during training, rather than a component of the proposed ZO algorithm.

Calibration protocol. Only WIS requires activation calibration. We obtain 128 examples from a separate iterator over the task training split, corresponding to 16 mini-batches of size 8. The forward passes are executed under inference mode, and the mask uses only the accumulated input activation norms. Although the standard teacher-forcing batches contain labels, neither label contents, loss values, nor gradients enter mask construction. Calibration examples and later training mini-batches are sampled from the same training split and may overlap. W-small uses only weight magnitudes, while Random requires no calibration.

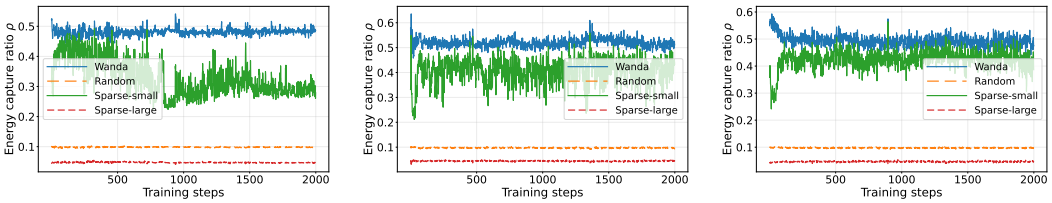
All methods use the same row-wise density ratio r, model parameters, gradient snapshots, and evaluation data. Equation equation 12 defines the empirical capture ratio for each mini-batch gradient snapshot. Its 2,000-step average is used for quantitative comparison.

For uniform row-wise random masking, each coordinate is selected with probability r. Conditioning on any fixed gradient snapshot $\mathbf{G}^t$, the expected captured energy over the random mask is

$$\mathbb{E}_{\mathbf{M}_{\text{random}}}\left[\rho_t(\mathbf{M}_{\text{random}}) \mid \mathbf{G}^t\right] = r. \tag{13}$$

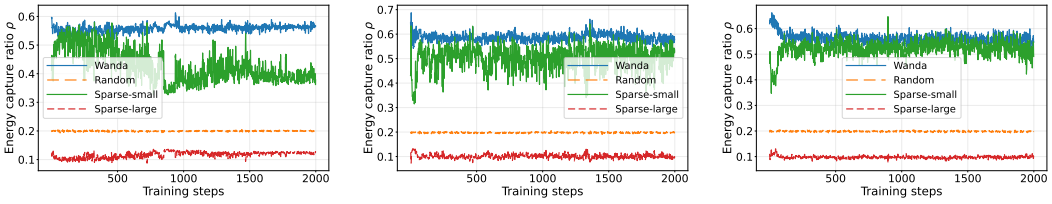
The main text reports the results at $r = 0.05$, where WIS consistently preserves more than 40% of the gradient energy and substantially outperforms Random, W-small, and W-large. W-small also captures more gradient energy than random masking, whereas W-large can fall below the random baseline. These results suggest that gradient energy is not positively correlated with parameter magnitude. In particular, the stronger energy capture of W-small relative to W-large indicates that small-magnitude parameters can carry substantial gradient signals, providing empirical support for the small-magnitude perturbation strategy used by Sparse-MeZO.

Figures 3–5 report the additional results at density ratios of 0.1, 0.2, and 0.4, respectively.



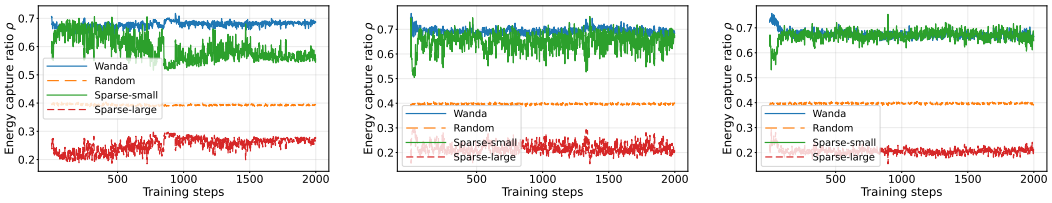
(a) RTE (b) QQP (c) MNLI

Figure 3: **Gradient energy capture ratio ρ on LLaMA-3.2-3B at a density ratio of $r = 0.1$.** Random denotes uniform random masking. W-small and W-large select coordinates with the smallest and largest weight magnitudes, respectively, while WIS selects coordinates using Wanda weight–activation scores.



(a) RTE (b) QQP (c) MNLI

Figure 4: **Gradient energy capture ratio ρ on LLaMA-3.2-3B at a density ratio of $r = 0.2$.** All mask construction strategies use the same row-wise perturbation density, model parameters, gradient snapshots, and evaluation data.



(a) RTE (b) QQP (c) MNLI

Figure 5: **Gradient energy capture ratio ρ on LLaMA-3.2-3B at a density ratio of $r = 0.4$.** Random denotes uniform random masking. W-small and W-large select coordinates with the smallest and largest weight magnitudes, respectively, while WIS uses Wanda weight–activation scores.

Across the evaluated tasks and density ratios, WIS achieves the highest gradient energy capture in nearly all settings. Its advantage is particularly pronounced at low density, where the perturbation budget is limited and support selection is most consequential. WIS also generally outperforms W-small, showing that task-dependent activation statistics provide complementary information beyond the small-magnitude criterion used by Sparse-MeZO. The only exception is MNLI at $r = 0.4$, where WIS and W-small achieve comparable energy capture. This convergence at a relatively high density is expected, as larger supports reduce the sensitivity to the precise ranking criterion and increase the overlap among different masks.

W-small consistently captures more gradient energy than W-large and generally outperforms uniform random masking. This finding provides empirical support for perturbing small-magnitude parameters, as adopted by Sparse-MeZO, and indicates that the observed gradient energy is not monotonically increasing with weight magnitude in the evaluated settings. However, the overall advantage of WIS demonstrates that weight magnitude alone is insufficient to identify the most informative perturbation coordinates. By incorporating downstream activation statistics, WIS constructs a task-conditioned support that more closely aligns with the gradient energy distribution.

Table 8: Peak GPU memory (GB, batch size=16) and iteration speed (iteration/s) for fine-tuning Qwen2.5-3B.

Methods	SST-2		ReCoRD		CB	
	Mem.	Iter/s	Mem.	Iter/s	Mem.	Iter/s
FT	35.5	3.37	87.4	0.92	94.5	0.75
LoRA	22.1	4.73	73.5	1.33	85.0	0.94
MeZO	9.70	3.50	21.5	1.39	45.1	0.99
Sparse-MeZO	16.2	2.98	27.3	1.30	51.0	0.95
WIS-MeZO	11.6	3.16	25.1	1.33	47.0	0.97

Table 9: **End-to-end wall-clock time (hours) for fine-tuning Qwen2.5-3B.** FT and LoRA are omitted because they use different training budgets and are therefore not directly comparable.

Methods	SST-2	ReCoRD	CB
MeZO	1.59	4.00	5.61
Sparse-MeZO	1.86	4.27	5.85
WIS-MeZO	1.76	4.18	5.73

C.2 Memory and Speed Comparison

We compare the memory footprint and computational efficiency of WIS-MeZO with MeZO, Sparse-MeZO, full-parameter fine-tuning (FT), and LoRA. All measurements are conducted on Qwen2.5-3B with a batch size of 16. Table 8 reports the peak GPU memory usage and iteration throughput, while Table 9 reports the end-to-end wall-clock time of the ZO methods. FT and LoRA are excluded from the wall-clock comparison because they use different training budgets and are therefore not directly comparable. We further evaluate the standard and 1-bit mask implementations of WIS-MeZO on Qwen2.5-32B, as summarized in Table 10.

On the 3B model, WIS-MeZO consistently uses less peak memory and achieves higher throughput than the evaluated Sparse-MeZO implementation. Relative to MeZO, the element-wise mask operations introduce a modest per-step overhead, which becomes less pronounced on tasks with longer forward passes. This trend is also reflected in the end-to-end wall-clock results. WIS-MeZO requires 1.76, 4.18, and 5.73 hours on SST-2, ReCoRD, and CB, respectively, corresponding to overheads of approximately 10.7%, 4.5%, and 2.1% over MeZO. Nevertheless, WIS-MeZO remains faster than Sparse-MeZO on all three tasks, reducing its wall-clock time by approximately 2.0%–5.4%. These results show that the additional task-conditioned mask operations introduce only moderate computational cost, particularly when the forward-pass computation dominates the iteration time.

On the 32B model, standard WIS-MeZO incurs approximately 25–28GB of additional memory relative to MeZO, primarily due to Boolean mask storage. Bit-packing the mask in WIS-MeZO (1-bit) reduces this overhead to less than 3GB, bringing its memory consumption close to MeZO. Across both ARC-C and HellaSwag, the measured throughput of the standard and 1-bit implementations also remains close to that of MeZO, indicating that the mask representation can substantially reduce memory overhead without introducing a large throughput penalty.

Table 10: Peak GPU memory (GB, batch size=16) and iteration speed (iteration/s) for fine-tuning Qwen2.5-32B.

Methods	ARC-C		HellaSwag	
	Mem.	Iter/s	Mem.	Iter/s
FT	600.5	0.126	664.8	0.074
LoRA	256.0	0.161	282.2	0.105
MeZO	97.29	0.174	104.8	0.104
WIS-MeZO	122.0	0.169	132.3	0.102
WIS-MeZO (1-bit)	98.84	0.160	107.7	0.099

C.3 Study on Density Ratio-r

WIS-MeZO in Algorithm 1 incorporates a key hyperparameter, the density ratio r, which governs the density (or sparsity level) of the perturbation mask. Here, we provide a detailed sensitivity analysis regarding this parameter. Table 11 presents the experimental results across a range of values for r.

Table 11: The effects of density ratio-r for Fine-tuning Qwen2.5-3B with WIS-MeZO.

Ratio-r	ARC-E	ARC-C	HellaSwag	WinoGrande	OpenBookQA
0.05	85.0	57.8	71.8	68.6	**54.0**
0.1	**85.4**	**60.8**	**73.5**	**69.7**	52.8
0.2	83.8	59.5	72.0	69.5	53.6
0.4	84.2	59.8	71.4	67.1	52.8
MeZO	82.9	57.8	69.9	66.3	49.0

As observed from the results, across all tested density ratios, WIS-MeZO consistently outperforms the MeZO baseline, further demonstrating the efficacy of our proposed method. Within the specified range of the ratio hyperparameter, the performance gains of WIS-MeZO remain robust. Furthermore, the optimal performance is generally achieved at a ratio of 0.1. In practice, $r = 0.1$ serves as a recommended starting point for hyperparameter tuning.

C.4 Effects of Calibration Size and Computation Cost

We study the sensitivity of WIS-MeZO to the number of calibration samples and the associated mask construction cost. As shown in Table 12, increasing the calibration size from 32 or 64 to 128 generally improves downstream performance, whereas further increasing it to 256 or 512 yields only marginal changes. The average scores for calibration sizes of 128, 256, and 512 are 68.86, 68.84, and 68.96, respectively, indicating that the mask quality largely stabilizes at 128 samples. This observation is consistent with the calibration-size analysis in Wanda (Sun et al., 2024).

Table 13 reports the corresponding mask construction time. The computation cost increases with the calibration size and also varies with the input characteristics of each task. Nevertheless, at the default calibration size of 128, mask construction requires only 0.53–1.26 seconds across the five benchmarks. Since fixed-mask WIS-MeZO performs this computation only once before fine-tuning, its overall overhead is negligible relative to the full optimization process. We therefore use 128 calibration samples by default, providing a favorable trade-off between mask quality and computation cost.

Table 12: Effects of the calibration size on WIS-MeZO fine-tuning of Qwen2.5-3B.

Calibration size	ARC-E	ARC-C	HellaSwag	WinoGrande	OpenBookQA
32	85.0	60.5	74.0	67.4	54.0
64	85.2	60.5	74.3	67.8	53.0
128 (default)	85.4	60.8	74.4	69.7	54.0
256	85.4	60.5	74.6	69.7	54.0
512	85.9	60.8	74.5	69.7	53.9

Table 13: WIS mask construction time in seconds for Qwen2.5-3B under different calibration sizes.

Calibration size	ARC-E	ARC-C	HellaSwag	WinoGrande	OpenBookQA
32	0.32	0.33	0.43	0.26	0.27
64	0.49	0.49	0.72	0.34	0.38
128 (default)	0.80	0.89	1.26	0.53	0.60
256	1.49	1.55	2.36	0.91	1.00
512	2.70	2.98	4.59	1.66	1.84

Algorithm 2 Perturb Parameters

Function $\mathrm{PerturbParameters}(\mathbf{W}, r, \epsilon, \mathbf{M}, s)$:
Input: Weights $\mathbf{W} \in \mathbb{R}^{m\times n}$, density ratio r, perturbation scale ϵ, mask $\mathbf{M} \in \{0,1\}^{m\times n}$, and random seed s
Reset the random number generator with seed s
Sample $\mathbf{Z} \in \mathbb{R}^{m\times n}$ with i.i.d. standard Gaussian entries
$\mathbf{W} \leftarrow \mathbf{W} + \frac{\epsilon}{\sqrt{r}}\mathbf{Z} \odot \mathbf{M}$ {In-place perturbation}
return $\mathbf{W}$

Algorithm 3 Compute Mask

Function $\mathrm{ComputeMask}(\mathbf{W}, \mathbf{X}, r)$:
Input: Weights $\mathbf{W} \in \mathbb{R}^{m\times n}$, calibration activations $\mathbf{X} \in \mathbb{R}^{n\times B}$, and density ratio r
$q_j \leftarrow \|\mathbf{X}_{j,:}\|_2$, for $j = 1, \ldots, n$
$\mathbf{S}_{i,j} \leftarrow |\mathbf{W}_{i,j}| q_j$
$k \leftarrow \lfloor nr \rfloor$ {Selected parameters per row}
for each row i in $\mathbf{S}$ **do**
  $\mathcal{I}_i \leftarrow \mathrm{TopK}(\mathbf{S}_{i,:}, k)$
  $\mathbf{M}_{i,j} \leftarrow \mathbb{I}[j \in \mathcal{I}_i]$, for $j = 1, \ldots, n$
end for
return $\mathbf{M}$

D Detailed Pseudocode

This section provides the detailed implementations of the parameter perturbation and WIS mask construction routines used in Algorithm 1. For each linear layer, $\mathbf{W} \in \mathbb{R}^{m\times n}$ denotes its weight matrix, $\mathbf{Z} \in \mathbb{R}^{m\times n}$ denotes the random perturbation, and $\mathbf{M} \in \{0,1\}^{m\times n}$ denotes the corresponding binary mask.

D.1 Layer-Wise Parameter Perturbation

In the implementation, Algorithm 2 is applied sequentially to the parameter matrices of all selected linear layers. The layers are traversed in a fixed deterministic order whenever the random seed is replayed. Therefore, the same perturbation is regenerated for the positive evaluation, negative evaluation, parameter recovery, and final update.

Each perturbation matrix $\mathbf{Z}$ is generated and released layer by layer. Consequently, WIS-MeZO does not store a full-model perturbation tensor, an additional perturbed model copy, or a full-model ZO gradient. The persistent storage overhead relative to MeZO is primarily the binary mask $\mathbf{M}$.

D.2 WIS Mask Construction

For each input feature j, Algorithm 3 first computes $q_j = \|\mathbf{X}_{j,:}\|_2$ from the calibration activations. The Wanda importance score of weight $\mathbf{W}_{i,j}$ is then

$$\mathbf{S}_{i,j} = |\mathbf{W}_{i,j}| q_j.$$

For every output row, the $k = \lfloor nr \rfloor$ coordinates with the largest scores are selected. This row-wise selection ensures that each output feature uses the same perturbation density r.

The calibration samples are drawn from the training split of the corresponding downstream task. Following Wanda, they are used only in forward passes to collect input-feature statistics. Labels and gradients are not required for mask construction. We use $B = 128$ calibration samples by default.

For fixed-mask WIS-MeZO, Algorithm 3 is executed once at the beginning of training. WISD-MeZO optionally repeats the same procedure every τ optimization steps.

E PROMPTS

In this section, we present the evaluation prompts employed throughout our study in Table 14. Our evaluation prompts follow MeZO (Malladi et al., 2023) and previous literature (Brown et al., 2020; Bach et al., 2022), with minor adaptations for our specific tasks.

Table 14: Prompt templates used in our experiments. cls. denotes classification with label-word verbalization, mch. denotes multiple-choice continuation scoring, and QA denotes generative question answering. <text> represents input from the dataset

Dataset	Type	Prompt
SST-2	cls.	<text> It was Labels: terrible/great
RTE	cls.	<premise> Does this mean that "<hypothesis>" is true? Yes or No? Labels: Yes/No
CB	cls.	Suppose <premise> Can we infer that "<hypothesis>"? Yes, No, or Maybe? Labels: Yes/No/Maybe
BoolQ	cls.	<passage> Question: <question> Labels: Yes/No
WSC	cls.	<text> In the previous sentence, does the pronoun "<span2>" refer to <span1>? Yes or No? Labels: Yes/No
WiC	cls.	Does the word "<word>" have the same meaning in these two sentences? Yes, No? <sent1> <sent2> Labels: Yes/No
MultiRC	cls.	<paragraph> Question: <question> I found this answer "<answer>". Is that correct? Yes or No? Labels: Yes/No
COPA	mch.	<premise> so/because <candidate>
WinoGrande	mch.	Prefix-before-blank continuation scoring: prefix(<sentence with _>) Candidate continuation: <option + suffix>
HellaSwag	mch.	<ctx> Answer: <candidate ending>
PIQA	mch.	Goal: <goal> Answer: <candidate solution>
ARC-E/ARC-C	mch.	Question: <question> Answer: <candidate choice text>
OpenBookQA	mch.	Question: <question_stem> Answer: <candidate choice text>
ReCoRD	mch.	<passage> <query>.replace("@placeholder", <candidate>)
SQuAD	QA	Title: <title> Context: <context> Question: <question> Answer:
DROP	QA	Passage: <context> Question: <question> Answer: