GA-LoRA: A Gradient-Free Path to Robust Fine-Tuning

Anonymous authors
Paper under double-blind review

Abstract

Fine-tuning a pre-trained model on a target distribution can improve in-distribution accuracy at the cost of the out-of-distribution robustness acquired during pre-training. Parameter-efficient fine-tuning methods such as Low-Rank Adaptation (LoRA) mitigate this tension by restricting updates to a compact subspace, yet the choice of optimizer still governs the geometry of the resulting solution. We introduce GA-LoRA, a gradient-free fine-tuning framework that evolves LoRA parameters with a Genetic Algorithm (GA), directly maximizing task accuracy as its fitness objective without backpropagation. We show analytically that the mutation–selection dynamics of GA-LoRA preferentially retain solutions in broader basins characterized by lower average-case sharpness. Empirically, across multiple model architectures and distribution-shift benchmarks, GA-LoRA preserves competitive in-distribution accuracy while consistently improving out-of-distribution generalization relative to gradient-based optimizers. These results establish evolutionary optimization as a principled and effective pathway toward robust fine-tuning of large-scale pre-trained models.

1 Introduction

Large-scale Pre-Trained Models (PTMs) have fundamentally advanced representation learning, exhibiting remarkable zero-shot generalization across diverse domains (Radford et al., 2021b). Despite their strength as general-purpose backbones, PTMs often underperform fully supervised models when directly applied to specialized downstream tasks (Wortsman et al., 2022). However, adapting PTMs through full fine-tuning frequently compromises the broad robustness acquired during pre-training, leading to substantial performance degradation under out-of-distribution (OOD) shifts (Wortsman et al., 2022; Pham et al., 2023). This tension between task-specific adaptation and preservation of generalization capability limits the reliable deployment of PTMs in real-world scenarios where data distributions are unpredictable (Kim et al., 2024).

Recent advances in Parameter-Efficient Fine-Tuning (PEFT) have demonstrated that training only a small number of additional or selected parameters can often effectively adapt PTMs to downstream tasks (Houlsby et al., 2019b; Lester et al., 2021; Li & Liang, 2021b). Among PEFT approaches, Low-Rank Adaptation (LoRA) (Hu et al., 2022) has achieved particularly strong results on vision-language PTMs such as CLIP (Radford et al., 2021a) and BLIP (Jia et al., 2021), yielding competitive in-distribution (ID) and OOD generalization (Zanella & Ayed, 2024; Yang et al., 2024). Nevertheless, LoRA still relies on gradient-based optimization to update its low-rank matrices, and such optimizers are prone to converging toward sharp minima that have been widely associated with degraded generalization and heightened sensitivity to distribution shifts (Belgoumri et al., 2025; Jin et al., 2025). Therefore, improving PEFT robustness from the perspective of the optimization algorithm remains an important yet underexplored direction.

An alternative perspective is offered by gradient-free optimization, which searches the parameter space without relying on backpropagated gradients (Leung et al., 2003; Esfahanian & Akhavan, 2019). Within this domain, evolutionary algorithms, particularly Genetic Algorithms (GAs) (Bäck, 1996; Ashlock, 2006), are widely utilized. By maintaining a diverse population of candidate solutions and performing population-level exploration, GAs are less susceptible to convergence toward sharp minima and are potentially well-suited for discovering flatter regions with improved generalization

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.

properties (Kitano, 1990). However, applying evolutionary optimization directly to PTM fine-tuning remains impractical due to the high dimensionality of modern model parameters.

Motivated by the well-established connection between flat minima and improved generalization, we investigate whether evolutionary optimization can steer PEFT toward broader low-loss regions. We analyze the mutation–selection dynamics of GAs and show that, under local assumptions, solutions residing in wider basins are more likely to produce viable offspring under stochastic mutation and consequently accumulate population mass through selection. This provides a geometric motivation for combining gradient-free evolutionary optimization with PEFT.

Building on this insight, we propose **GA-LoRA**, a gradient-free fine-tuning framework that integrates GAs with LoRA parameterization. The low-dimensional parameterization of LoRA alleviates the curse of dimensionality that has traditionally limited evolutionary optimization of large-scale PTMs, rendering evolutionary optimization practical for model adaptation. Extensive experiments across multiple downstream benchmarks and model architectures demonstrate that GA-LoRA consistently improves OOD generalization while maintaining competitive ID performance. We further examine the resulting loss geometry using both worst-case and average-case sharpness under multiple perturbation radii and find that GA-LoRA converges to wider, flatter regions than gradient-based LoRA, corroborating the proposed mutation–selection mechanism and providing a geometric explanation for its improved robustness. Our contributions are summarized as follows:

- We identify and exploit the synergy between PEFT and gradient-free optimization: the low-rank parameterization of LoRA reduces the dimensionality barrier that limits evolutionary optimization, enabling efficient gradient-free adaptation of large-scale PTMs.
- We propose GA-LoRA, a novel gradient-free fine-tuning framework that replaces gradient-based updates with evolutionary optimization over LoRA parameters, improving OOD generalization while preserving efficiency and ID accuracy.
- We provide a comprehensive geometric analysis of the optimization landscape, demonstrating that GA-LoRA converges to wider, flatter minima than gradient-based LoRA. This reveals a mechanism by which evolutionary optimization improves robustness and offers new insights into robust PEFT.

2 Related Work

2.1 Out-of-Distribution (OOD) Generalization

OOD generalization, also referred to as natural distribution robustness, measures a model's resilience to real-world distributional shifts such as variations in scene composition, object categories, and illumination (Taori et al., 2020).Large-scale PTMs, including CLIP (Radford et al., 2021a) and BLIP (Jia et al., 2021), exhibit strong robustness under such shifts, yet unrestricted fine-tuning on task-specific data can erode the general-purpose representations learned during pre-training (Kumar et al., 2022; Pham et al., 2023), particularly in few-shot regimes where models are prone to representation collapse (Wang et al., 2025; Guo et al., 2025). PEFT offers a natural mitigation strategy by freezing most of the PTM and updating only a small set of task-specific parameters (Houlsby et al., 2019b; Lester et al., 2021; Li & Liang, 2021b). Existing PEFT methods include bias-only tuning (Ben Zaken et al., 2022), sparse or selective parameter updates (Zhao et al., 2020; Sung et al., 2021; Xu et al., 2021), adapter-based tuning (Houlsby et al., 2019a), prompt-based tuning (Li & Liang, 2021a; Liu et al., 2021), and LoRA (Zanella & Ayed, 2024). By restricting the trainable subspace, these methods reduce representation drift and better preserve the transferability of the PTMs, while the optimizer determines which solution is selected within that subspace. This motivates investigating whether the choice of optimization strategy for PEFT can further improve OOD generalization.

2.2 Loss Geometry and Generalization

The geometry of learned solutions has long been studied as an explanation for differences in generalization. Large-batch training has been associated with convergence to sharp minima and a corresponding generalization gap (Keskar et al., 2017). Several methods explicitly incorporate neighborhood information: Entropy-SGD biases optimization toward wide valleys (Chaudhari et al., 2017), stochastic weight averaging favors wider optima through parameter averaging (Izmailov et al., 2019), and SAM optimizes a local worst-case perturbation objective (Foret et al., 2021). More recent

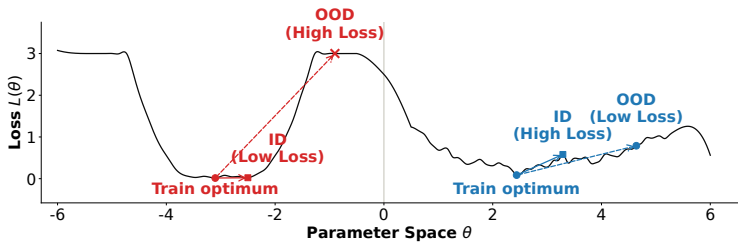


Figure 1: **Illustration of loss geometry and OOD generalization.** Solutions with similar training objectives may exhibit different local basin geometries and OOD behavior, motivating the search for broader low-loss regions.

empirical studies show that the relationship between sharpness and generalization depends on the perturbation definition, parameterization, and evaluation protocol (Andriushchenko et al., 2023). Motivated by these observations, we analyze GA-LoRA using two complementary sharpness metrics across multiple perturbation radii to provide a more comprehensive characterization of the local loss geometry. Figure 1 illustrates the geometric intuition underlying this work.

2.3 Genetic Algorithms (GAs)

Genetic algorithms optimize a population of candidates through selection, crossover, and mutation without requiring gradient information (Leung et al., 2003; Esfahanian & Akhavan, 2019). This makes them well-suited for discontinuous or black-box objectives and for search spaces where gradient information is unreliable. In machine learning, GAs have been applied to hyperparameter optimization and neural architecture search (Aszemi & Dominic, 2019; Alibrahim & Ludwig, 2021; Sun et al., 2019), as well as direct neural-network training (Leung et al., 2003; Esfahanian & Akhavan, 2019; Such et al., 2018; Sanchez Carmona et al., 2024). However, GA efficiency degrades significantly as the search dimensionality increases: compared with gradient-based methods, GAs struggle with the fine-grained local refinement required in later stages of convergence (Kitano, 1990). This scalability limitation is particularly acute for PTMs, where evaluating large populations of high-dimensional parameter vectors is computationally prohibitive. This work integrates GAs into the PEFT framework, substantially reducing the search space and making GA-based optimization tractable, thereby helping to bridge classical GAs and modern PTM adaptation.

3 Method

We present a unified formulation that combines LoRA with evolutionary optimization. The key idea is to treat the LoRA parameters as a population of chromosomes and evolve them through selection, crossover, and mutation without computing gradients. Population-level search allows mutation-tolerant, high-fitness regions to be preferentially retained, while the compact LoRA parameterization keeps the GA-based optimization tractable.

3.1 Problem Formulation and Chromosome Encoding

Low-Rank Adaptation. Fine-tuning all parameters of a large-scale PTM is computationally expensive. LoRA (Hu et al., 2022) exploits the observation that weight updates often reside in a low-dimensional subspace. Instead of modifying a full weight matrix $W \in \mathbb{R}^{d\times k}$, LoRA parameterizes the update as

$$\Delta W = BA, \tag{1}$$

where $A \in \mathbb{R}^{r\times k}$ and $B \in \mathbb{R}^{d\times r}$ with rank $r \ll \min(d, k)$. During training, only the low-rank matrices A and B are learned; the original weights remain frozen.

Model and objective. We instantiate GA-LoRA on CLIP, a dual-encoder vision-language model. Let $f_{\text{img}}(\cdot)$ and $f_{\text{text}}(\cdot)$ denote the visual and textual encoders, each equipped with LoRA adapters.

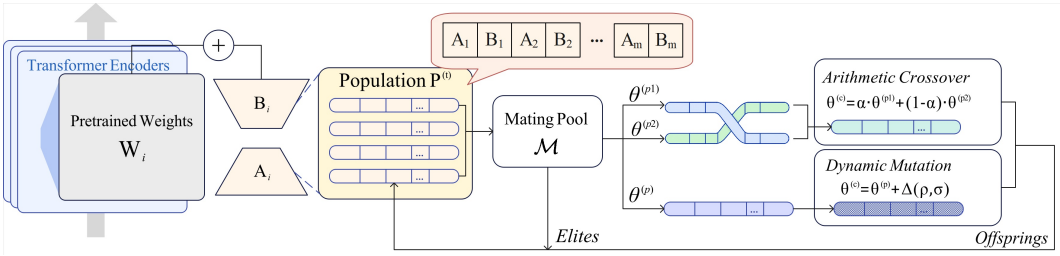


Figure 2: Overview of the GA-LoRA optimization framework.

For an image x and a set of K candidate classes $\{c_k\}_{k=1}^{K}$, we compute normalized embeddings:

$$\hat{\mathbf{v}}(x) = \frac{f_{\text{img}}(x)}{\|f_{\text{img}}(x)\|_2}, \qquad \hat{\mathbf{t}}(c_k) = \frac{f_{\text{text}}(p(c_k))}{\|f_{\text{text}}(p(c_k))\|_2}, \tag{2}$$

where $p(\cdot)$ is a prompt template. Prediction follows zero-shot classification:

$$\hat{y} = \arg\max_k \ \hat{\mathbf{v}}(x)^\top \hat{\mathbf{t}}(c_k). \tag{3}$$

Let the model contain M LoRA-adapted layers. For the i-th adapted layer, we introduce matrices $A_i \in \mathbb{R}^{d\times r}$ and $B_i \in \mathbb{R}^{r\times k}$ with rank $r \ll \min(d,k)$. The complete set of LoRA parameters,

$$\theta = \{A_1, B_1, \ldots, A_M, B_M\}, \tag{4}$$

constitutes the **chromosome** of an individual in our evolutionary algorithm. Our sole objective is to maximize the top-1 accuracy on an ID training set $\mathcal{D} = \{(x_n, y_n)\}_{n=1}^{N}$:

$$\mathcal{F}(\theta) = \frac{1}{N}\sum_{n=1}^{N} \mathbb{1}\big[\hat{y}_n(\theta) = y_n\big]. \tag{5}$$

No regularization term is added; any improvement in OOD generalization emerges purely from the properties of the evolutionary optimization.

3.2 GA-LoRA: Evolutionary Optimization of LoRA Parameters

GA-LoRA follows the standard generational loop illustrated in Figure 2 and detailed in Algorithm 1. An initial population of P individuals is evaluated using Equation (5). In each generation, elite individuals are preserved, and parents are selected to produce offspring through crossover and mutation. Because fitness evaluation requires only forward passes, the framework is entirely gradient-free and trivially parallelizable.

Initialization and elitism. Each low-rank matrix A_i is initialized with Kaiming uniform values, while B_i is set to zero, so that every individual initially preserves the output of the frozen pre-trained model. To prevent degradation, the E individuals with the highest fitness are directly copied into the next generation (elitism). A mating pool $\mathcal{M}$ is then formed by selecting the S fittest individuals, offering a tunable balance between selection pressure and population diversity.

Arithmetic crossover. With probability P_{cross}, two parents are recombined via convex combination:

$$\theta^{(c)} = \alpha \cdot \theta^{(p1)} + (1-\alpha)\cdot\theta^{(p2)}, \tag{6}$$

where $\alpha \sim U(0,1)$ is sampled independently for each crossover event. Because LoRA parameters lie in a continuous space, arithmetic crossover naturally interpolates between different adaptation directions, combining beneficial parameter configurations discovered by separate individuals. This is consistent with prior work showing that parameter interpolation can improve robustness by avoiding over-specialized solutions (Izmailov et al., 2019; Wortsman et al., 2022).

Dynamic mutation with annealing. If crossover is not selected, a single parent undergoes mutation. We apply masked isotropic Gaussian noise to every weight matrix $W \in \theta$:

$$W' = W + \mathbf{M} \odot \mathbf{N}, \quad \mathbf{N}_{ij} \sim \mathcal{N}(0, \sigma^2), \ \ m_{ij} \sim \text{Bernoulli}(\rho). \tag{7}$$

Algorithm 1 GA-LoRA: Genetic Optimization of LoRA Parameters

Input: training dataset $\mathcal{D}$, frozen CLIP model with LoRA layers
Output: optimised LoRA parameters $\theta^\star$
Initialize population $\mathcal{P} = \{\theta^{(1)}, \ldots, \theta^{(P)}\}$ and evaluate $\mathcal{F}(\theta)$ for all $\theta \in \mathcal{P}$
for $g = 1$ **to** G **do**
  {Elites} $\mathcal{E} \leftarrow$ top E individuals by fitness
  {Mating pool} $\mathcal{M} \leftarrow$ top S individuals by fitness
  $\mathcal{P}' \leftarrow \mathcal{E}$
  while $|\mathcal{P}'| < P$ **do**
    Sample parents $\theta^{(p1)}, \theta^{(p2)} \sim \mathcal{M}$
    Draw $u \sim U(0, 1)$
    if $u < P_{\text{cross}}$ **then**
      $\theta^{(c)} \leftarrow \text{ArithmeticCrossover}(\theta^{(p1)}, \theta^{(p2)})$
    else
      $\theta^{(c)} \leftarrow \text{DynamicMutation}(\theta^{(p1)})$
    end if
    $\mathcal{P}' \leftarrow \mathcal{P}' \cup \{\theta^{(c)}\}$
  end while
  $\mathcal{P} \leftarrow \mathcal{P}'$ and re-evaluate $\mathcal{F}(\theta)$ for all $\theta \in \mathcal{P}$
end for
$\theta^\star \leftarrow \arg\max_{\theta \in \mathcal{P}} \mathcal{F}(\theta)$
return $\theta^\star$

The masked Gaussian mutation enables efficient exploration of the continuous LoRA parameter space while preserving useful adaptation structures. The mutation probability ρ and strength σ are gradually reduced over G generations:

$$\rho^{(g)} = \rho_{\max} - \frac{g}{G}(\rho_{\max} - \rho_{\min}), \tag{8}$$

$$\sigma^{(g)} = \sigma_{\max} - \frac{g}{G}(\sigma_{\max} - \sigma_{\min}). \tag{9}$$

Early-stage large perturbations encourage broad exploration and prevent premature convergence, while later-stage smaller perturbations enable fine-grained refinement around promising solutions (Such et al., 2018).

3.3 Mutation–Selection Mechanism

Since GA-LoRA does not explicitly impose any sharpness constraint during evolution, the mutation–selection mechanism alone cannot yield a strict generalization bound based on sharpness minimization. Instead, our analysis provides a local intuition for why evolutionary optimization may favor solutions residing in broader low-loss regions.

The underlying mechanism is as follows: mutation introduces stochastic perturbations around current solutions, and selection preserves individuals whose offspring remain within low-loss regions. As shown in the local analysis of Appendix C, the probability of generating viable offspring depends on the local curvature of the loss landscape. When two solutions have comparable loss and similar basin structure, the solution in the broader region is more likely to retain low-loss offspring after mutation, gradually increasing its share of the population through selection.

Figure 3 provides a synthetic illustration of this mutation–selection process, demonstrating how a population initially distributed across different low-loss basins can progressively concentrate in the wider basin as evolutionary selection favors individuals with higher offspring viability.

4 Experiments

4.1 Experimental Setup

Baselines. We compare GA-LoRA against the following optimization algorithms:

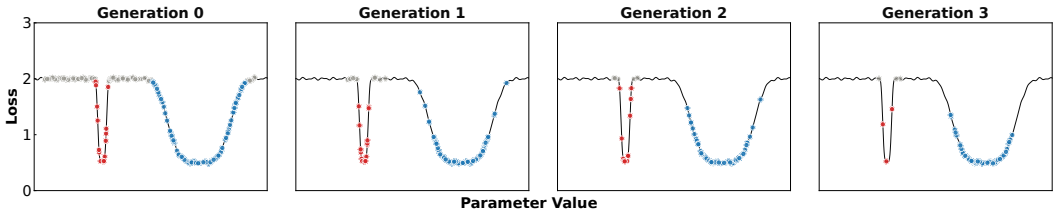


Figure 3: **Simulated GA evolution in a one-dimensional parameter space.** Both basins have identical minimum loss, yet the population is progressively drawn into the wider basin.

- **Stochastic Gradient Descent (SGD)** performs iterative updates using stochastic gradients computed from mini-batches:
$$\theta_{t+1} = \theta_t - \eta \nabla_\theta \mathcal{L}(\theta_t). \tag{10}$$
We include both vanilla SGD and SGD with momentum (SGD-M).
- **Adaptive Moment Estimation (Adam)** adapts per-parameter learning rates using exponential moving averages of the first and second moments of the gradient:
$$\theta_{t+1} = \theta_t - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}. \tag{11}$$
- **Entropy-SGD (E-SGD)** biases optimization toward flat minima by maximizing a local-entropy objective, using nested Stochastic Gradient Langevin Dynamics (SGLD) loops (Chaudhari et al., 2017):
$$\theta_{t+1} = \theta_t - \eta\gamma(\theta_t - \mu), \tag{12}$$
where μ is the exponential average of parameters explored via L inner-loop SGLD iterations.
- **Sharpness-Aware Minimization (SAM)** explicitly seeks flat minima by optimizing for the worst-case perturbation ϵ within a neighborhood of θ (Foret et al., 2021):
$$\min_\theta \max_{\|\epsilon\|_2 \le \rho} \mathcal{L}(\theta + \epsilon). \tag{13}$$
We also adopt its scale-invariant variant AdaptiveSAM (ASAM) (Kwon et al., 2021).

Datasets. We train on the ImageNet training set and evaluate on its validation set for ID performance. For OOD generalization, we use five benchmarks exhibiting substantial distribution shifts: ImageNet-V2 (Recht et al., 2019), ImageNet-Sketch (Wang et al., 2019), ImageNet-R (Hendrycks et al., 2021a), ImageNet-A (Hendrycks et al., 2021b), and ObjectNet (Barbu et al., 2019).

Metrics. We primarily report the average accuracy across the five OOD datasets to evaluate natural distribution robustness. We additionally adopt *Effective Robustness (ER)* (Taori et al., 2020) as a complementary metric. ER exploits the strong linear correlation between ID and OOD accuracy observed among fine-tuned models (e.g., ImageNet vs. ImageNet-V2). For a model f, let $acc_1(f)$ and $acc_2(f)$ denote its ID and OOD accuracies. ER is defined as
$$\rho(f) = acc_2(f) - \beta(acc_1(f)), \tag{14}$$
where $\beta(acc_1(f))$ is the OOD accuracy predicted by the linear relationship. A positive $\rho(f)$ indicates robustness beyond the expected ID–OOD trade-off, providing a measure of distribution-shift robustness disentangled from ID accuracy.

Implementation details. We apply LoRA with rank $r = 2$ (Ghiasvand et al., 2025; Zanella & Ayed, 2024) to the query, key, and value projection matrices of the vision encoder. For a fair comparison, all hyperparameters for both GA-LoRA and the baselines are tuned extensively, with final configurations selected based on the highest ID accuracy on the ImageNet training set.

For all gradient-based methods, we employ a batch size of 64 and train for $500 \cdot \frac{N}{K}$ iterations, with N being the total labeled samples and K the number of classes. We conduct a grid search for learning rates η within the range $[10^{-5}, 10^{-1}]$. Specifically, we set $\eta = 0.001$ for Adam and $\eta = 0.01$ for SGD, SGD-M, and SAM. For Entropy-SGD, we use $L = 20$ inner-loop SGLD iterations with a scoping parameter $\gamma = 0.01$. For SAM, the neighborhood size is set to $\rho = 0.05$.

GA-LoRA maintains a population of $P = 100$ with $E = 4$ elites. We use $G = 500$ generations. For exploration, the mutation strength σ and sparsity ρ are linearly annealed from 0.01 to 5×10^{-4} and 1.0 to 0.05, respectively. For crossover, we set $P_{cross} = 0.3$ and $\alpha = 0.5$. We provide an ablation study for mutation operators in Appendix A.

Table 1: ID and OOD accuracy (%) across different shot regimes using ViT-B/16 as backbone. "Gap" denotes the generalization gap (ID − OOD_{Avg}). Best results are **bolded** and second-best are underlined. Results are averaged over 3 random seeds.

Shots	Optimizer	ID	OOD						Gap	ER
		IN	Avg.	IN-V2	IN-S	IN-A	IN-R	Obj	(Low ↓)	(High ↑)
0	Zero-shot	66.70	57.07	60.81	46.09	46.81	74.47	57.17	9.63	4.67
16	SGD	69.59	59.58	63.16	48.29	48.72	77.30	59.56	10.01	4.50
	SGD-M	70.84	59.28	64.12	47.98	48.58	76.65	59.05	11.56	4.33
	Adam	70.52	59.73	63.87	48.58	48.52	77.62	60.09	10.79	4.44
	E-SGD	70.42	60.08	64.00	49.01	48.32	**78.08**	**61.02**	10.34	4.62
	SAM	70.91	59.56	64.29	48.56	48.12	77.62	59.63	11.51	4.34
	ASAM	**71.07**	59.64	64.35	48.49	48.12	77.60	59.63	11.51	4.30
	GA (Ours)	70.88	**61.41**	**65.23**	**50.92**	**51.23**	**77.88**	**61.81**	**9.27**	**4.68**
4	SGD	69.05	59.02	62.64	47.98	47.73	76.88	59.87	10.03	4.45
	SGD-M	69.57	59.43	63.26	48.22	48.98	77.00	59.67	10.14	4.53
	Adam	68.84	58.10	61.96	46.82	49.05	75.10	57.56	10.74	3.96
	E-SGD	69.58	60.41	63.22	48.64	50.12	77.30	62.76	9.17	4.62
	SAM	69.58	58.69	62.78	47.49	48.65	76.32	58.20	10.89	4.20
	ASAM	69.61	58.57	61.78	47.68	48.62	76.53	58.25	11.04	4.15
	GA (Ours)	**69.63**	**61.38**	**64.73**	**49.24**	**51.05**	**78.38**	**63.52**	**8.25**	**4.66**
1	SGD	67.25	57.36	61.08	46.36	47.00	74.78	57.56	9.89	**4.67**
	SGD-M	67.71	57.87	61.43	46.85	47.20	75.70	58.17	9.84	4.42
	Adam	**68.56**	58.87	61.56	48.01	47.47	76.77	**60.56**	9.69	3.80
	E-SGD	68.38	58.78	**62.07**	47.50	48.19	76.73	59.45	9.60	4.47
	SAM	68.49	58.86	61.63	48.01	47.33	76.78	60.52	9.63	3.87
	ASAM	**68.56**	58.85	61.73	48.15	47.33	76.59	60.45	9.71	3.85
	GA (Ours)	68.48	**59.33**	62.04	**48.26**	**49.02**	**76.88**	60.45	**9.15**	**4.67**

4.2 Main Results

As shown in Table 1, GA-LoRA consistently achieves superior OOD generalization while maintaining ER close to that of the zero-shot model. Although SAM and ASAM slightly exceed GA-LoRA in ID accuracy in some settings, this advantage comes at the cost of OOD generalization. For example, under the 16-shot setting, ASAM attains the highest ID accuracy (71.07%), yet its OOD average (59.64%) falls notably below that of GA-LoRA (61.41%). This trade-off underscores a key advantage of evolutionary optimization: it effectively balances ID performance with strong OOD generalization.

4.3 Performance vs. Training Data Size and LoRA Rank

We further examine optimizer robustness under varying training data sizes and LoRA ranks. As shown in Figure 4 (right), ID accuracy consistently improves as the number of training shots increases from 1 to 64, indicating that more labeled samples facilitate better ID fitting. In contrast, ER scales much more weakly with training data size and even declines for some gradient-based optimizers in larger-shot regimes. GA-LoRA exhibits the most stable robustness trend, maintaining high ER across all shot settings with minimal variation.

Table 2 compares different LoRA ranks under the 16-shot setting. While increasing the rank does not consistently improve ID accuracy, GA-LoRA achieves the best OOD average across all ranks, suggesting that its robustness advantage is not tied to a specific LoRA rank configuration.

4.4 Cross-Architecture Generalization

We evaluate whether GA-LoRA generalizes across different architectures and modalities. Table 3 summarizes results on multiple CLIP vision encoders, the CLIP text encoder, the BLIP-2 Q-Former, and GPT-2 language models. Across these diverse settings, GA-LoRA consistently achieves the strongest OOD generalization while maintaining competitive ID accuracy. For vision-language models, GA-LoRA yields consistent OOD improvements across different CLIP variants and the BLIP-2 Q-Former. Beyond vision-language models, we adapt GPT-2-Small and GPT-2-Large with

Table 3: **Generalization across different architectures and modalities.** Values are accuracy (%). Best ID and OOD results in each block are bold.

Method	CLIP B/16		CLIP B/32		CLIP L/14		CLIP Text		BLIP-2		GPT-2 Small		GPT-2 Large	
	ID	OOD	ID	OOD	ID	OOD	ID	OOD	ID	OOD	ID	OOD	ID	OOD
SGD	69.6	59.6	64.7	49.6	76.3	71.8	70.2	59.9	74.8	64.5	70.5	76.6	84.6	91.6
Adam	70.5	59.7	64.8	48.7	77.8	72.3	71.3	59.8	74.8	63.3	71.3	76.1	**85.5**	91.3
E-SGD	70.4	60.1	65.1	50.1	76.7	72.8	70.4	61.5	74.9	65.1	71.0	78.0	85.1	92.2
ASAM	**71.4**	60.0	**65.7**	48.9	76.4	70.1	**71.9**	60.0	**76.2**	64.6	**71.9**	76.8	85.4	91.9
GA	70.8	**60.5**	65.3	**52.1**	**76.9**	**74.4**	71.7	**62.7**	75.9	**66.9**	71.5	**79.3**	85.4	**92.9**

LoRA on SST-2, using SST-2 validation accuracy as ID performance and evaluating distribution shifts on IMDb and Yelp Polarity. GA-LoRA achieves the highest OOD accuracy for both GPT-2 variants. These results suggest that the advantages of GA-LoRA generalize across different PTMs.

5 Sharpness Analysis

Section 3.3 predicts a preference for mutation-tolerant neighborhoods but does not imply that GA-LoRA minimizes every notion of sharpness. To test this prediction and disentangle mutation tolerance from sharpness minimization, we therefore employ two complementary diagnostics, described below.

5.1 Metrics

For a perturbation radius $\rho > 0$, **adaptive worst-case sharpness** $S^{\rho}_{\max}(w)$ and **adaptive average-case sharpness** $S^{\rho}_{\text{avg}}(w)$ are defined as

$$S^{\rho}_{\max}(w) = \max_{\|\epsilon \oslash |w|\|_{\infty} \le \rho} \left[\mathcal{L}(w+\epsilon) - \mathcal{L}(w)\right], \tag{15}$$

$$S^{\rho}_{\text{avg}}(w) = \mathbb{E}_{\epsilon \sim \mathcal{N}(0,\Sigma)}[\mathcal{L}(w+\epsilon)] - \mathcal{L}(w), \qquad \Sigma_{ii} = (\rho|w_i|)^2. \tag{16}$$

$S^{\rho}_{\max}$ captures worst-case sensitivity to adversarial perturbations which is aligned with SAM's objective. S^{ρ}_{avg} captures average-case sensitivity to random weight-scaled perturbations, which is aligned with the random perturbations used by GA. Detailed procedures are given in Appendix D.

5.2 Empirical Verification

Figure 5(a–b) shows that the two metrics rank optimizers differently. SAM generally achieves the lowest $S_{\max}$, as expected from its adversarial neighborhood objective. GA-LoRA instead tends to exhibit lower S_{avg} at larger radii, indicating a smaller expected loss increase under typical adaptive mutations. This is consistent with the mutation–selection mechanism.

The relationships between sharpness and ER shown in Figure 5(c–d) reveal how different sharpness measures relate to generalization. Among the evaluated checkpoints, $S_{\max}$ exhibits only a weak association with ER ($r \approx 0.27$), whereas S_{avg} shows a stronger negative correlation ($r \approx -0.69$). These results suggest that average-case mutation sensitivity is more predictive of the robustness

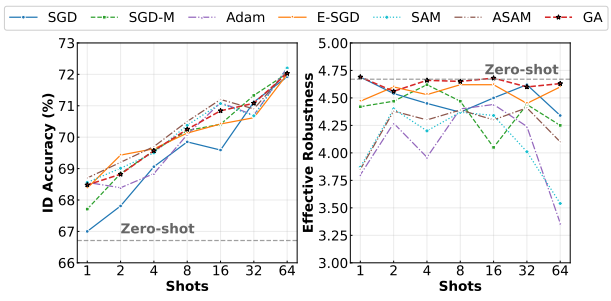


Figure 4: ID accuracy and ER under different shot settings.

Table 2: Cross-rank comparison

Method	$r=2$		$r=4$		$r=8$	
	ID	OOD	ID	OOD	ID	OOD
SGD	69.6	59.6	70.4	59.8	70.0	59.6
Adam	70.5	59.7	70.4	58.5	70.0	58.1
E-SGD	70.4	60.1	70.3	60.9	69.8	59.4
SAM	71.1	59.6	70.3	58.5	**70.2**	58.0
ASAM	**71.4**	60.0	71.1	60.0	70.1	59.7
GA	70.8	**60.5**	**71.4**	**61.3**	70.1	**60.8**

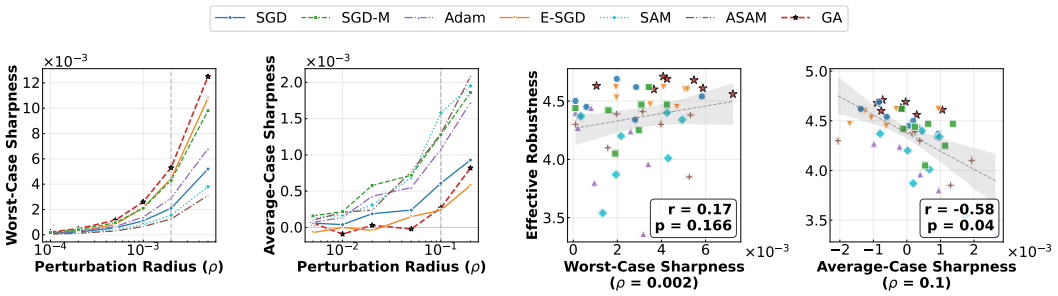


Figure 5: **Sharpness diagnostics.** (a–b) Adaptive worst-case and average-case sharpness across perturbation radii. (c–d) Their association with ER.

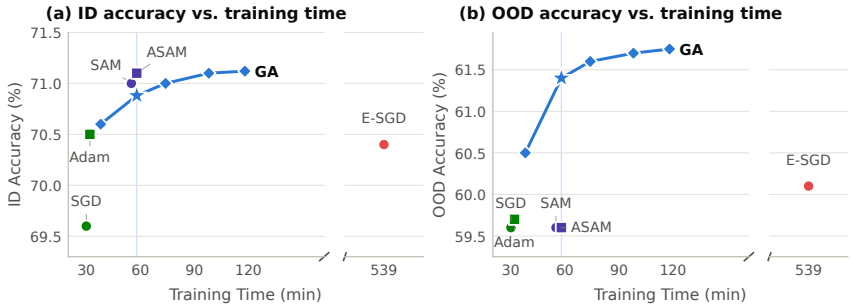


Figure 6: **Robustness–efficiency trade-off.** (a) ID accuracy and (b) OOD accuracy versus wall-clock training time. The starred GA point marks the training budget adopted in the main experiments.

improvements achieved by GA-LoRA, while worst-case sharpness alone does not fully capture the optimizer-induced generalization differences.

6 Robustness–Efficiency Trade-off

Figure 6 presents the ID and OOD accuracy trajectories as the GA generation budget increases. With a limited evolution budget, OOD improvements are modest, while larger gains emerge with extended evolutionary optimization. This reflects the inherent exploration cost of gradient-free optimization and highlights the trade-off between computational expenditure and robustness improvement. For reference, Entropy-SGD also incurs significant computational overhead due to its inner SGLD loops, resulting in higher training costs than standard gradient-based optimizers.

The fitness evaluations of different candidates in GA-LoRA are independent and can be distributed across GPUs, and early stopping eliminates many low-fitness candidates before a full evaluation pass. These properties make the computational cost tunable, yielding a trade-off between resource consumption and robustness. Appendix B analyzes the computational complexity of each optimizer.

7 Conclusion

We introduced GA-LoRA, a gradient-free optimization framework for PEFT that combines the restricted update space of LoRA with evolutionary mutation–selection dynamics. GA-LoRA preserves competitive ID accuracy while favoring mutation-tolerant neighborhoods in the adapted parameter space. Across experiments on CLIP, BLIP-2, and GPT-2, GA-LoRA consistently improves OOD generalization, maintaining ER close to the zero-shot reference. The sharpness analysis reveals an association with lower average-case mutation sensitivity and demonstrates that worst-case and average-case sharpness capture complementary aspects of the solution geometry.

Limitations. The current implementation incurs higher single-GPU training time than standard gradient-based optimizers, and the GPT-2 experiments cover only two model sizes and a single sentiment-analysis protocol. Scaling to larger language models, multimodal LLMs, and a broader set of tasks remains future work.

AI USE STATEMENT

In this work, generative AI tools were utilized exclusively for assisting with initial language proofreading and LaTeX template formatting. All core conceptual ideas, theoretical derivations, algorithm designs, code implementations, and experimental formulations were independently conceived, verified, and authored by the researchers. The authors take full responsibility for the entirety of the content, claims, and formulations presented in this paper.

ETHICS STATEMENT

This research focuses on foundational machine learning algorithms. All datasets used are standard, publicly available academic benchmarks. The proposed method does not generate harmful content, involve sensitive personal data, or introduce societal bias.

REPRODUCIBILITY STATEMENT

To ensure reproducibility, comprehensive implementation details are provided in Section 4.1.

REFERENCES

Hussain Alibrahim and Simone A Ludwig. Hyperparameter optimization: Comparing genetic algorithm against grid search and bayesian optimization. In *2021 IEEE congress on evolutionary computation (CEC)*, pp. 1551–1559. IEEE, 2021.

Maksym Andriushchenko, Francesco Croce, Maximilian Müller, Matthias Hein, and Nicolas Flammarion. A modern look at the relationship between sharpness and generalization. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 840–902, 2023.

Daniel Ashlock. *Evolutionary Computation for Modeling and Optimization*. Springer Science & Business Media, 2006. ISBN 9780387221964.

Nurshazlyn Mohd Aszemi and P Dhanapal Durai Dominic. Hyperparameter optimization in convolutional neural network using genetic algorithms. *International Journal of Advanced Computer Science and Applications*, 10(6), 2019.

Thomas Bäck. *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford University Press, 1996. ISBN 9780195099719.

Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. URL https://arxiv.org/abs/1812.06417.

Mohammed D Belgoumri, Mohamed Reda Bouadjenek, Hakim Hacid, Imran Razzak, and Sunil Aryal. Navigating loss manifolds via rigid body dynamics: A promising avenue for robustness and generalisation. *arXiv preprint arXiv:2505.19527*, 2025.

Elad Ben Zaken, Shauli Ravfogel, and Yoav Goldberg. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *ACL*, 2022.

Pratik Chaudhari, Anna Choromanska, Stefano Soatto, Yann LeCun, Carlo Baldassi, Christian Borgs, Jennifer Chayes, Levent Sagun, and Riccardo Zecchina. Entropy-sgd: Biasing gradient descent into wide valleys. In *International Conference on Learning Representations (ICLR)*, 2017. URL https://arxiv.org/abs/1611.01838v5. Code: https://github.com/ucla-vision/entropy-sgd.

Parsa Esfahanian and Mohammad Akhavan. GACNN: Training Deep Convolutional Neural Networks with Genetic Algorithm, September 2019. URL http://arxiv.org/abs/1909.13354. arXiv:1909.13354 [cs].

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-Aware Minimization for Efficiently Improving Generalization, April 2021. URL http://arxiv.org/abs/2010.01412. arXiv:2010.01412 [cs].

Sajjad Ghiasvand, Haniyeh Ehsani Oskouie, Mahnoosh Alizadeh, and Ramtin Pedarsani. Few-shot adversarial low-rank fine-tuning of vision-language models. *ArXiv*, abs/2505.15130, 2025.

Qianyu Guo, Jingrong Wu, Tianxing Wu, Haofen Wang, Weifeng Ge, and Wenqiang Zhang. Enhancing environmental robustness in few-shot learning via conditional representation learning. *IEEE Transactions on Image Processing*, 34:3489–3502, 2025.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *International Conference on Computer Vision (ICCV)*, 2021a. URL https://arxiv.org/abs/2006.16241.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021b. URL https://arxiv.org/abs/1907.07174.

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 *ICML*, 2019a.

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 *International conference on machine learning*, pp. 2790–2799. PMLR, 2019b.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging Weights Leads to Wider Optima and Better Generalization, February 2019. URL http://arxiv.org/abs/1803.05407. arXiv:1803.05407 [cs].

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision, 2021. URL https://arxiv.org/abs/2102.05918.

Long Jin, Han Nong, Liangming Chen, and Zhenming Su. A method for enhancing generalization of adam by multiple integrations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 4147–4155, 2025.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations*, 2017.

Sungyeon Kim, Boseung Jeong, Donghyun Kim, and Suha Kwak. Efficient and versatile robust fine-tuning of zero-shot models. In *European Conference on Computer Vision*, pp. 440–458. Springer, 2024.

Hiroaki Kitano. 1990-Empirical Studies on the Speed of Convergence of Neural Network Training Using Genetic Algorithms. 1990.

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations*, 2022.

Jungmin Kwon, Jeongseop Kim, Hyunseo Park, and In Kwon Choi. Asam: Adaptive sharpness-aware minimization for scale-invariant learning of deep neural networks. In *International conference on machine learning*, pp. 5905–5914. PMLR, 2021.

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, 2021.

Frank Hung-Fat Leung, Hak-Keung Lam, Sai-Ho Ling, and Peter Kwong-Shun Tam. Tuning of the structure and parameters of a neural network using an improved genetic algorithm. *IEEE Transactions on Neural networks*, 14(1):79–88, 2003.

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

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing (volume 1: Long papers)*, pp. 4582–4597, 2021b.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. Gpt understands, too. In *arXiv:2103.10385*, 2021.

Hieu Pham, Zihang Dai, Golnaz Ghiasi, Kenji Kawaguchi, Hanxiao Liu, Adams Wei Yu, Jiahui Yu, Yi-Ting Chen, Minh-Thang Luong, Yonghui Wu, Mingxing Tan, and Quoc V. Le. Combined scaling for zero-shot transfer learning, 2023. URL https://arxiv.org/abs/2111.10050.

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, 2021a. URL https://arxiv.org/abs/2103.00020.

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, February 2021b. URL http://arxiv.org/abs/2103.00020. arXiv:2103.00020 [cs].

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning (ICML)*, 2019. URL https://arxiv.org/abs/1902.10811.

Vicente Ivan Sanchez Carmona, Shanshan Jiang, and Bin Dong. How Well Can a Genetic Algorithm Fine-tune Transformer Encoders? A First Approach. In *Proceedings of the Fifth Workshop on Insights from Negative Results in NLP*, pp. 25–33, Mexico City, Mexico, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.insights-1.4. URL https://aclanthology.org/2024.insights-1.4.

Felipe Petroski Such, Vashisht Madhavan, Edoardo Conti, Joel Lehman, Kenneth O. Stanley, and Jeff Clune. Deep Neuroevolution: Genetic Algorithms Are a Competitive Alternative for Training Deep Neural Networks for Reinforcement Learning, April 2018. URL http://arxiv.org/abs/1712.06567. arXiv:1712.06567 [cs].

Yanan Sun, Bing Xue, Mengjie Zhang, and Gary G. Yen. Evolving deep convolutional neural networks for image classification, 2019. URL http://arxiv.org/abs/1710.10741.

Yi-Lin Sung, Varun Nair, and Colin Raffel. Training neural networks with fixed sparse masks. In *NeurIPS*, 2021.

Rohan Taori, Achal Dave, Vaishaal Shankar, and Nicholas Carlini. Measuring Robustness to Natural Distribution Shifts in Image Classification. 2020.

Hanxin Wang, Tian Liu, and Shu Kong. Robust Few-Shot Vision-Language Model Adaptation, June 2025. URL http://arxiv.org/abs/2506.04713. arXiv:2506.04713 [cs].

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. URL https://arxiv.org/abs/1905.13549.

Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, and Ludwig Schmidt. Robust fine-tuning of zero-shot models. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7949–7961, New Orleans, LA, USA, June 2022. IEEE. ISBN 978-1-6654-6946-3. doi: 10.1109/CVPR52688.2022.00780. URL https://ieeexplore.ieee.org/document/9878622/.

Runxin Xu, Fuli Luo, Zhiyuan Zhang, Chuanqi Tan, Baobao Chang, Songfang Huang, and Fei Huang. Raise a child in large language model: Towards effective and generalizable fine-tuning. In *EMNLP*, 2021.

Menglin Yang, Jialin Chen, Yifei Zhang, Jiahong Liu, Jiasheng Zhang, Qiyao Ma, Harshit Verma, Qianru Zhang, Min Zhou, Irwin King, and Rex Ying. Low-Rank Adaptation for Foundation Models: A Comprehensive Review, December 2024. URL http://arxiv.org/abs/2501.00365. arXiv:2501.00365 [cs].

Maxime Zanella and Ismail Ben Ayed. Low-Rank Few-Shot Adaptation of Vision-Language Models, June 2024. URL http://arxiv.org/abs/2405.18541. arXiv:2405.18541 [cs].

Mengjie Zhao, Tao Lin, Martin Jaggi, and Hinrich Schütze. Masking as an efficient alternative to finetuning for pretrained language models. In *EMNLP*, 2020.

A Ablation Study

We investigate the influence of the crossover probability (P_{cross}) and the final mutation rate (ρ_{end}) on the 8-shot ImageNet benchmark. As shown in Table 4, we report training, ID, and OOD accuracies after 500 generations. The results indicate that a hybrid strategy with $P_{\text{cross}} = 0.3$, combined with a moderate final mutation rate of $\rho_{\text{end}} = 0.05$, achieves the best trade-off between exploration and exploitation, yielding the highest performance across all evaluation metrics.

Table 4: **Ablation study on 8-shot ImageNet.** Comparison of Training, ID, and Average OOD accuracy after 500 generations.

Component	Setting	Train (%)	ID (%)	OOD (%)
Crossover P_{cross}	0 (*Mutation Only*)	71.6	68.5	58.2
	0.3	**73.8**	**70.3**	**60.8**
	0.5	73.4	69.8	59.9
	0.7	72.2	69.2	59.1
	1.0	69.9	67.5	57.5
Mutation ρ_{end}	1.0	70.1	67.1	58.4
	0.1	72.9	69.7	60.1
	0.05	**73.8**	**70.3**	**60.8**
	0.01	73.3	70.1	60.2

B Computational Complexity

We analyze the theoretical computational complexity of GA-LoRA relative to the gradient-based baselines.

Let C_F denote the computational cost of a forward pass and C_B the cost of a backward pass. We define the following variables based on our experimental setup:

- B: Batch size.
- T_{grad}: Total iterations for gradient methods.
- L: Number of inner Langevin dynamics steps for Entropy-SGD.
- G: Number of generations for GA.
- P: Population size.
- N_{eff}: Effective number of samples evaluated per individual. We adopt dynamic *Early Stopping* during fitness evaluation: individuals whose estimated performance cannot exceed the current best fitness are terminated early. In our main experiments, $N_{eff} = 100$, which substantially reduces evaluation cost while maintaining reliable fitness estimation.

The computational complexity (in terms of FLOPs proportional to forward pass units) for each algorithm is estimated as follows:

- **SGD / Adam:** Requires one forward and one backward pass per iteration.

$$\text{Cost}_{\text{SGD}} \approx T_{grad} \times B \times (C_F + C_B) \approx 3 \cdot T_{grad} \cdot B \cdot C_F \tag{17}$$

- **SAM:** Requires two forward-backward passes per step (one to compute ϵ, one to update).

$$\text{Cost}_{\text{SAM}} \approx T_{grad} \times B \times 2(C_F + C_B) \approx 6 \cdot T_{grad} \cdot B \cdot C_F \tag{18}$$

- **Entropy-SGD:** Requires L inner steps of SGLD for every outer update

$$\text{Cost}_{\text{Entropy}} \approx T_{grad} \times B \times L \times (C_F + C_B) \approx 3L \cdot T_{grad} \cdot B \cdot C_F \tag{19}$$

- **GA:** Requires only forward passes for fitness evaluation ($C_B = 0$).

$$\text{Cost}_{\text{GA}} \approx G \times P \times N_{eff} \times C_F \tag{20}$$

C Formal Analysis of Mutation–Selection Dynamics

We formalize the local mutation–selection mechanism described in Section 3.3. The key intuition is that individuals in sharp basins are less robust to mutation: their offspring are more likely to land in high-loss regions and thus fail to survive selection. In contrast, individuals in flatter basins produce viable high-fitness offspring with higher probability, causing population mass to gradually concentrate around such regions.

C.1 Setup

Let $\mathcal{L} : U \subset \mathbb{R}^d \to \mathbb{R}$ be a C^4 empirical loss function. Consider a finite set of isolated local minima

$$w_1, \ldots, w_K.$$

For a parent parameter vector w, we model mutation as an isotropic Gaussian perturbation whose scale is proportional to the magnitude of the parent:

$$\epsilon_w \sim \mathcal{N}(0, \Sigma(w)), \qquad \Sigma(w) = \rho^2 m(w)^2 I_d, \qquad m(w) := \|w\|_\infty,$$

or equivalently,

$$\epsilon_w = \rho\, m(w) z, \qquad z \sim \mathcal{N}(0, I_d).$$

For each local minimum w_j, define the scaled Hessian

$$A_j := m(w_j)^2 \nabla^2 \mathcal{L}(w_j).$$

Given a tolerance parameter $\tau > 0$, we define the viability probability of basin j as

$$P_j^\rho(\tau) := \mathbb{P}\left[\mathcal{L}(w_j + \epsilon_{w_j}) \le \mathcal{L}(w_j) + \frac{\rho^2}{2}\tau\right].$$

This quantity measures the probability that a mutated offspring remains within a low-loss neighborhood of its parent. We also define the average mutation-induced sharpness:

$$S_j^\rho := \mathbb{E}\left[\mathcal{L}(w_j + \epsilon_{w_j})\right] - \mathcal{L}(w_j).$$

C.2 Proposition

Proposition. Under the setup above, the following statements hold.

(1) Local mutation behavior. For each local minimum w_j,

$$P_j^\rho(\tau) \to \mathbb{P}\left[z^\top A_j z \le \tau\right] \qquad (\rho \to 0),$$

and

$$S_j^\rho = \frac{\rho^2}{2}\operatorname{Tr}(A_j) + O(\rho^4) = \frac{\rho^2}{2} m(w_j)^2 \operatorname{Tr}\big(\nabla^2 \mathcal{L}(w_j)\big) + O(\rho^4).$$

(2) Mean-field population update. Let $p_t(j)$ denote the fraction of the population located in basin j at generation t. In the mean-field limit, selection based on offspring viability induces the update

$$p_{t+1}(j) = \frac{p_t(j) P_j^\rho(\tau)}{\sum_{k=1}^K p_t(k) P_k^\rho(\tau)}.$$

Consequently, for any two basins j and k,

$$\frac{p_{t+1}(j)}{p_{t+1}(k)} = \frac{P_j^\rho(\tau)}{P_k^\rho(\tau)} \frac{p_t(j)}{p_t(k)}.$$

Thus, basins with larger viability probability gain relative population mass.

(3) Flatter minima have larger viability probability. If two basins f and s satisfy

$$A_f \preceq A_s,$$

then

$$\mathbb{P}\left[z^\top A_f z \le \tau\right] \ge \mathbb{P}\left[z^\top A_s z \le \tau\right],$$

and hence

$$P_f^\rho(\tau) \geq P_s^\rho(\tau) - o(1) \qquad (\rho \to 0).$$

Therefore, flatter minima generate viable high-fitness offspring more frequently.

Moreover, if the local basin shapes satisfy

$$A_j = \alpha_j A_0, \qquad A_0 \succeq 0,$$

for a shared shape matrix A_0, then

$$P_j^\rho(\tau) = \Psi_\tau(S_j^\rho) + o(1),$$

where Ψ_τ is a decreasing function. Hence, smaller average mutation-induced sharpness S_j^ρ implies larger viability probability and therefore larger asymptotic basin occupancy.

C.3 Proof

Fix a local minimum w_j and denote

$$H_j := \nabla^2 \mathcal{L}(w_j), \qquad A_j := m(w_j)^2 H_j.$$

Since w_j is a local minimum, we have

$$\nabla \mathcal{L}(w_j) = 0.$$

By Taylor expansion around w_j,

$$\mathcal{L}(w_j + \epsilon_{w_j}) - \mathcal{L}(w_j) = \frac{1}{2}\epsilon_{w_j}^\top H_j \epsilon_{w_j} + \frac{1}{6}\nabla^3 \mathcal{L}(w_j)[\epsilon_{w_j}, \epsilon_{w_j}, \epsilon_{w_j}] + O(\|\epsilon_{w_j}\|^4).$$

Using

$$\epsilon_{w_j} = \rho m(w_j) z, \qquad z \sim \mathcal{N}(0, I_d),$$

we obtain

$$\mathcal{L}(w_j + \epsilon_{w_j}) - \mathcal{L}(w_j) = \frac{\rho^2}{2} z^\top A_j z + O(\rho^3 \|z\|^3).$$

Therefore,

$$\frac{2}{\rho^2}\left(\mathcal{L}(w_j + \epsilon_{w_j}) - \mathcal{L}(w_j)\right) \to z^\top A_j z \qquad \text{a.s.}$$

As a result, for every continuity point τ of the distribution of $z^\top A_j z$,

$$P_j^\rho(\tau) = \mathbb{P}\left[\frac{2}{\rho^2}\left(\mathcal{L}(w_j + \epsilon_{w_j}) - \mathcal{L}(w_j)\right) \leq \tau\right] \to \mathbb{P}\left[z^\top A_j z \leq \tau\right].$$

Next, taking expectation in the Taylor expansion gives

$$S_j^\rho = \frac{1}{2}\mathbb{E}\left[\epsilon_{w_j}^\top H_j \epsilon_{w_j}\right] + \frac{1}{6}\mathbb{E}\left[\nabla^3 \mathcal{L}(w_j)[\epsilon_{w_j}, \epsilon_{w_j}, \epsilon_{w_j}]\right] + O(\rho^4).$$

Since ϵ_{w_j} is a zero-mean Gaussian vector, all third-order moments vanish, so

$$\mathbb{E}\left[\nabla^3 \mathcal{L}(w_j)[\epsilon_{w_j}, \epsilon_{w_j}, \epsilon_{w_j}]\right] = 0.$$

Thus,

$$S_j^\rho = \frac{1}{2}\operatorname{Tr}\left(H_j \Sigma(w_j)\right) + O(\rho^4).$$

Substituting

$$\Sigma(w_j) = \rho^2 m(w_j)^2 I_d,$$

we obtain

$$S_j^\rho = \frac{\rho^2}{2} m(w_j)^2 \operatorname{Tr}(H_j) + O(\rho^4) = \frac{\rho^2}{2}\operatorname{Tr}(A_j) + O(\rho^4).$$

This proves the first statement.

We now consider the population dynamics. Suppose that at generation t, a fraction $p_t(j)$ of the population lies in basin j. A parent from basin j produces a viable offspring with probability proportional to $P_j^\rho(\tau)$. After selection, the population mass assigned to basin j is therefore proportional to

$$p_t(j)P_j^\rho(\tau).$$

Normalizing across all basins gives the mean-field update

$$p_{t+1}(j) = \frac{p_t(j)P_j^\rho(\tau)}{\sum_{k=1}^K p_t(k)P_k^\rho(\tau)}.$$

Taking the ratio between two basins j and k, we obtain

$$\frac{p_{t+1}(j)}{p_{t+1}(k)} = \frac{P_j^\rho(\tau)}{P_k^\rho(\tau)}\frac{p_t(j)}{p_t(k)}.$$

Hence, basins with larger viability probability increase their relative population mass over generations.

Finally, suppose that two basins f and s satisfy

$$A_f \preceq A_s.$$

Then for every $z \in \mathbb{R}^d$,

$$z^\top A_f z \le z^\top A_s z.$$

Therefore,

$$\mathbf{1}_{\{z^\top A_f z \le \tau\}} \ge \mathbf{1}_{\{z^\top A_s z \le \tau\}}.$$

Taking expectations gives

$$\mathbb{P}\left[z^\top A_f z \le \tau\right] \ge \mathbb{P}\left[z^\top A_s z \le \tau\right].$$

Combining this with the convergence of $P_j^\rho(\tau)$ yields

$$P_f^\rho(\tau) \ge P_s^\rho(\tau) - o(1).$$

Thus, flatter minima produce viable offspring more frequently.

For the final claim, suppose

$$A_j = \alpha_j A_0, \qquad A_0 \succeq 0.$$

Then

$$z^\top A_j z = \alpha_j z^\top A_0 z,$$

and hence

$$P_j^\rho(\tau) \to \mathbb{P}\left[z^\top A_0 z \le \frac{\tau}{\alpha_j}\right].$$

Since the right-hand side is decreasing in α_j, a larger curvature scale α_j implies a smaller viability probability.

Moreover,

$$S_j^\rho = \frac{\rho^2}{2}\operatorname{Tr}(A_j) + O(\rho^4) = \frac{\rho^2}{2}\alpha_j \operatorname{Tr}(A_0) + O(\rho^4).$$

Thus, S_j^ρ is increasing in α_j, while $P_j^\rho(\tau)$ is decreasing in α_j. Therefore, there exists a decreasing function Ψ_τ such that

$$P_j^\rho(\tau) = \Psi_\tau(S_j^\rho) + o(1).$$

This shows that smaller average mutation-induced sharpness corresponds to larger offspring viability probability.

Overall, the mutation-selection dynamics favor basins with higher viable-offspring probability. Under the stated local assumptions, flatter basins have a higher such probability and therefore receive more population mass. This provides a formal mechanism that is consistent with the behavior observed in GA-LoRA.

D SHARPNESS CALCULATION ALGORITHMS

We provide detailed pseudocode for the two sharpness metrics used in our analysis: adaptive worst-case sharpness ($S_{\max}$) and adaptive average-case sharpness (S_{avg}). To ensure scale invariance across layers and parameters, both metrics employ element-wise adaptive perturbations proportional to the parameter magnitudes.

D.1 ADAPTIVE WORST-CASE SHARPNESS (S_{max})

We approximate the worst-case sharpness using multi-step Projected Gradient Descent (PGD). The objective is to find the perturbation δ that maximizes the loss within an adaptive neighborhood defined by radius ρ.

Algorithm 2 Computation of Adaptive Worst-Case Sharpness (S_{max})

Input: Model parameters θ, Batch data $\mathcal{B} = \{(x, y)\}$, Neighborhood radius ρ, PGD steps T, Step size α
Output: Worst-case sharpness value S_{max}
Compute initial loss: $L_{base} \leftarrow \mathcal{L}(f(x; \theta), y)$
Initialize perturbation: $\delta \leftarrow \mathbf{0}$ (or random initialization within bounds)
for $t = 1$ **to** T **do**
    Compute gradients w.r.t perturbation:
    $g \leftarrow \nabla_\delta \mathcal{L}(f(x; \theta + \delta), y)$
    // Adaptive Gradient Ascent Step
    $\delta \leftarrow \delta + \alpha \cdot \frac{g}{\|g\|_2}$
    // Projection onto the adaptive constraint set
    // Constraint: $|\delta_i| \leq \rho|\theta_i|$
    $\delta \leftarrow \textbf{Clip}(\delta, -\rho|\theta|, \rho|\theta|)$
end for
Compute perturbed loss: $L_{perturbed} \leftarrow \mathcal{L}(f(x; \theta + \delta), y)$
$S_{max} \leftarrow L_{perturbed} - L_{base}$
return S_{max}

D.2 ADAPTIVE AVERAGE-CASE SHARPNESS (S_{avg})

Average-case sharpness estimates the expected loss increase under random Gaussian perturbations. We use Monte Carlo sampling to approximate the expectation. Crucially, the variance of the noise is scaled by the magnitude of the parameters to maintain adaptivity.

Algorithm 3 Computation of Adaptive Average-Case Sharpness (S_{avg})

Input: Model parameters θ, Batch data $\mathcal{B} = \{(x, y)\}$, Perturbation radius/strength ρ, Number of MC samples K
Output: Average-case sharpness value S_{avg}
Compute initial loss: $L_{base} \leftarrow \mathcal{L}(f(x; \theta), y)$
$L_{sum} \leftarrow 0$
for $k = 1$ **to** K **do**
    Sample standard Gaussian noise: $\xi \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
    // Adaptive scaling: noise is proportional to parameter magnitude
    $\delta \leftarrow \rho \cdot \xi \odot |\theta|$
    Compute loss with random perturbation:
    $L_k \leftarrow \mathcal{L}(f(x; \theta + \delta), y)$
    $L_{sum} \leftarrow L_{sum} + L_k$
end for
$L_{avg} \leftarrow L_{sum}/K$
$S_{avg} \leftarrow L_{avg} - L_{base}$
return S_{avg}