

**On Adversarial and Common Robustness of Parameter-Efficient  
Fine-Tuning Strategies**

by

Kunyang Li

A thesis submitted in partial fulfillment of  
the requirements for the degree of

Master of Science

(Computer Sciences)

at the

UNIVERSITY OF WISCONSIN–MADISON

2024

Date of thesis submission: 12/13/2024

The thesis is approved by the following members of the Committee:

Patrick D. McDaniel

Tsun-Ming Shih Professor of Computer Sciences, School of Computer,  
Data, & Information Sciences

Thesis Advisor

Kassem M. Fawaz

Associate Professor, Electrical and Computer Engineering, Computer Sci-  
ences (Affiliate)

Associate Chair for Research

Rahul Chatterjee

Assistant Professor, School of Computer, Data, & Information Sciences

© Copyright by Kunyang Li 2024  
All Rights Reserved

## CONTENTS

---

### Contents

List of Tables iii

List of Figures iv

Abstract vi

**1 Introduction** 1

**2 Background** 5

2.1 *Vision Transformer Architecture* 5

2.2 *Parameter-Efficient Fine-Tuning Strategies* 7

2.3 *Model Robustness* 10

**3 Methodology** 12

3.1 *Threat Model* 12

3.2 *The Space of PEFTs* 13

3.3 *Sensitivity Analysis for Robustness* 16

**4 Evaluation** 20

4.1 *Experimental Setup* 20

4.2 *Trade-off Between Accuracy and Robustness* 25

4.3 *Pareto Front Curves in the Trade-off Space* 29

4.4 *On Out-of-Distribution Robustness* 32

**5 Discussion & Limitations** 36

5.1 *Defining the Space of PEFTs* 36

5.2 *Adversarial Training* 38

5.3 *Security & Safety Measures* 38

<b>6</b>	<b>Related Work</b>	<b>39</b>
6.1	<i>Robustness of Traditional Transfer Learning</i>	39
6.2	<i>Hybrid PEFT Methods</i>	40
<b>7</b>	<b>Conclusion</b>	<b>41</b>
<b>8</b>	<b>Appendix</b>	<b>42</b>
8.1	<i>Training Curves with Adversarial Robustness</i>	42
8.2	<i>More Results on OOD Robustness</i>	46
	<b>Bibliography</b>	<b>51</b>

**LIST OF TABLES**

---

2.1	Categorization of state-of-the-art PEFT techniques . . . . .	7
3.1	The space of PEFT strategies in terms of information location and underlying mechanisms . . . . .	15
4.1	Configurations of PEFTs based on standard practices . . . . .	21
4.2	Strategy configurations with datasets (Adv) . . . . .	22
4.3	Strategy configurations with datasets (OOD) . . . . .	22
4.4	Datasets and robustness measures . . . . .	23
4.5	Area under the curve (AUC) of the Pareto front curves . . . . .	31

## LIST OF FIGURES

---

1.1	<b>Thesis summary table.</b> This table summarizes three main research questions investigated, the corresponding measurements, and simplified visualization for results. . . . .	3
2.1	<b>Vision Transformer Architecture.</b> This is an architecture visualization of one ViT block. . . . .	6
3.1	<b>A systematic framework for sensitivity analysis of PEFTs' robustness and accuracy.</b> A framework to characterize the trade-off space of PEFT strategies. (1) Select pre-trained models and downstream tasks; (2) Integrate PEFT modules to the pre-trained model; and (3) Dynamically evaluate their robustness and accuracy across backpropagations. . . . .	13
3.2	<b>Left:</b> A graphical illustration of how five different PEFTs are applied to one ViT block. <b>Right:</b> Individual PEFT mechanism.	14
3.3	<b>Tracking schedule for standard accuracy and Adv and OOD robustness.</b> Designed tracking schedule for efficiently attacking/evaluating model states during fine-tuning. . . . .	18
4.1	<b>PGD loss convergence plot during attacking.</b> It is one of the attack loss curves on CUB200 dataset with full fine-tuning in order to verify the loss is converging. . . . .	24
4.2	<b>The trend of training standard accuracy (blue), test standard accuracy (green), and adversarial robustness on test dataset (red) across the number of backpropagation steps on Caltech256.</b> PGD robustness reaches its peak and drops at an early stage of training, while both training and test clean accuracy keep increasing and plateau in the end. . . . .	26

4.3	<b>Trade-off visualization between standard accuracy and robustness for Caltech256.</b> The dots are corresponding to different time stamps during training (from bottom left to upper right to upper left as time goes on). . . . .	28
4.4	<b>The Pareto front curves of the trade-off between standard accuracy and robustness on Caltech256, CUB200, CIFAR10, and CIFAR100.</b> The Pareto front curves of different PEFT strategies reside in different locations in the trade-off space. . . . .	30
4.5	<b>The trend of training standard accuracy (blue), test standard accuracy in the training domain (green), and OOD robustness in other domains (red) across the number of backpropagation steps.</b> The results of two training domains—clip art and real images—from DomainNet [1] are shown here. . . . .	33
4.6	<b>A heatmap of the highest OOD robustness during training across 6 domains.</b> . . . . .	34
8.1	<b>The trend of training standard accuracy (blue), test standard accuracy (green), and adversarial robustness on test dataset (red) across the number of backpropagation steps on five datasets.</b> PGD robustness reaches its peak and drops at an early stage of training, while both training and test clean accuracy keep increasing and plateau in the end. . . . .	45
8.2	<b>The trend of training standard accuracy (blue), test standard accuracy in the training domain (green), and OOD robustness in other domains (red) across the number of backpropagation steps.</b> The results of six training domains are shown here. . . . .	49
8.3	<b>A heatmap of the converged OOD robustness towards the end of the fine-tuning phase across 6 domains.</b> . . . . .	50



## ABSTRACT

---

With the rise of Vision Transformer (ViT) architectures and large-scale online data, foundation models are first pre-trained on massive datasets to capture general knowledge and then fine-tuned on smaller and specific datasets for downstream tasks. Recently, parameter-efficient fine-tuning strategies (PEFTs) have emerged as a promising alternative to full-model fine-tuning, offering improved accuracy with reduced computational costs by strategically updating model parameters. However, the robustness of PEFTs, particularly in security and safety contexts—how the shifts to using PEFT strategies impact model reliability against adversarial attacks (security) and under unseen conditions (safety)—remains underexplored. This thesis bridges the gap between studies on robustness of traditional training and the emerging application of PEFTs with their limitations in practice. In this work, we characterize the robustness-accuracy trade-off space and its sensitivity to PEFT strategies, backpropagation steps, downstream tasks in the image domain. We propose a systematic framework that includes pre-training, fine-tuning, and robustness evaluation, integrating dynamic analysis and Pareto cost curves to assess model robustness. With this framework, we fine-tune 231 models with seven state-of-the-art fine-tuning methods across six datasets and perform  $\sim 2.1k$  adversarial security evaluations and  $\sim 2k$  out-of-distribution (OOD) safety evaluations. Our findings indicate that: (1) the adversarial robustness-accuracy trade-off consistently exhibits early in fine-tuning across PEFT strategies and downstream tasks; (2) the trade-off in security correlates strongly with downstream task complexity and similarity to the pre-training dataset as well as PEFT mechanisms; and (3) key features impacting robustness in security and safety are fundamentally different—there is no significant robustness-accuracy trade-off for OOD data, as the trend of OOD robustness aligns closely with that of standard accuracy during fine-tuning.

The thesis emphasizes the need for tailored techniques to preserve (for security) or enhance (for safety) robustness for PEFT strategies with minimal impact on achieving high standard accuracy, providing valuable insights for researchers and practitioners developing efficient and robust fine-tuning methods.

## 1 INTRODUCTION

---

The transformer architecture, introduced in 2017 [2], has become the state-of-the-art across many fields, including natural language processing (NLP) and computer vision (CV) [3, 4, 5]. It typically serves as the backbone architecture in the pre-training and fine-tuning paradigm [6] where general knowledge of pre-trained models is transferred to solve specific tasks through fine-tuning on (often small) downstream datasets. Given the high and exponentially growing computational, memory, and storage costs associated with fine-tuning an entire model, new parameter-efficient fine-tuning (PEFT) strategies [7, 8, 9, 10, 11, 12, 13, 14, 15, 16] have been replacing traditional fine-tuning methods (e.g., full fine-tuning and linear probing) by strategically inserting or selecting a small number of parameters (less than 5% of the pre-trained model) to be updated. They have become the de facto approach to achieve higher accuracy with less data and much lower computational requirements [7, 12, 17].

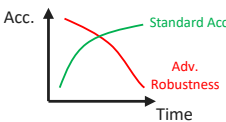
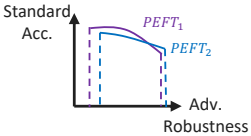
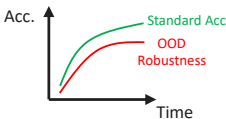
While these PEFTs aim to reduce computational overheads, they focus on improving accuracy on benign data. However, the trustworthiness of machine learning models has been a major concern in the field, especially when models are increasingly integrated into high-stakes applications such as healthcare [18], autonomous driving [19], and cybersecurity [20]. It is well-known that models are often vulnerable with previously-unseen phenomena (i.e., safety under out-of-distribution, OOD) and against adversarial inputs (i.e., security under intentionally crafted perturbations to cause model misclassification). Although tuning a small portion of pre-trained model weights can achieve high accuracy on downstream tasks [7, 12], it is unclear if PEFTs exacerbate the implicit vulnerabilities models have in the sense of security and safety. Furthermore, previous studies in traditional training settings (e.g., training from scratch, full fine-tuning, and linear probing) [21, 22, 23, 24] often stress the importance

of adversarial training, which adds generated adversarial examples to training data to improve robustness. However, since adversarial training requires full control of the pre-training process with expensive computational costs, most off-the-shelf pre-trained models in practice are standard-trained [25, 26, 27]. Thus, there is a *misalignment* between current studies on robustness and what developers are capable of performing today with parameter-efficient fine-tuning strategies.

Furthermore, the claim that adversarial examples are intrinsic features of *datasets* [22] has motivated our exploration of model robustness in the pre-training and fine-tuning paradigm. We hypothesize that a pre-trained model, trained on upstream datasets, is inherently more resistant to adversarial examples generated on downstream datasets. However, as various PEFT strategies adapt different parameters to downstream tasks, key questions arise: at what stages can the model fully exhibit robustness inherited from the pre-trained parameters, and to what extent does it start to learn “non-robust” features (i.e., highly predictive yet imperceptible) from downstream datasets during fine-tuning? Given the fundamental differences among accuracy, adversarial robustness, and OOD generalization robustness, we further hypothesize that these three metrics show distinct patterns at different stages of fine-tuning, each showing unique sensitivities to training time, PEFT mechanisms, and downstream tasks.

***Thesis Statement:*** *There exist distinct accuracy-robustness trade-offs in different parameter-efficient fine-tuning settings.*

In this work, we propose an approach to characterize the trade-off space of accuracy, security, and safety for PEFTs in the image domain. Our framework includes 1) pre-training, 2) fine-tuning, and 3) robustness evaluation. This provides flexibility for an in-depth, dynamic analysis of PEFT strategies and their influence on downstream models’ robustness throughout fine-tuning. We select SOTA PEFT strategies from three

Research Focus	Measure	Results
<b>Existence Proof:</b> Is there a robustness-accuracy trade-off during fine-tuning?	Accuracy/Robustness vs. <b>Time</b>	
<b>PEFTs Comparison:</b> Do different PEFT strategies offer different trade-off space?	AUC of Pareto front curves across <b>PEFT strategies</b>	
<b>On OOD:</b> Do results generalize to OOD settings?	Security $\rightarrow$ <b>Safety</b>	

**Figure 1.1: Thesis summary table.** This table summarizes three main research questions investigated, the corresponding measurements, and simplified visualization for results.

popular categories—insertion, selection, and reparameterization—to comprehensively explore the PEFT space. Robustness is evaluated in various settings, including adversarial attacks and distribution shifts, to capture a wide spectrum of challenges encountered in real-world applications.

In our experiments, we fine-tune 231 models using seven widely-used fine-tuning methods and evaluate their robustness on six common benchmarks with around 2.1k adversarial and 2k OOD robustness evaluations. Adversarial robustness was assessed using projected gradient descent (PGD) [28], a popular attack algorithm. Our findings reveal that: (1) across all fine-tuning strategies, model robustness initially increases with standard accuracy but begins to decline as accuracy continues to rise, highlighting an inherent trade-off between adversarial robustness and accuracy early in fine-tuning; (2) robustness in security contexts is sensitive to both PEFT mechanisms (i.e., each mechanism produces a distinct accuracy-robustness Pareto frontier) and downstream tasks (i.e., the trade-off is strongly correlated to the complexity of downstream tasks with respect to

the upstream datasets); and (3) robustness in security and safety contexts exhibits distinct characteristics during fine-tuning, with OOD robustness aligning closely with standard accuracy without any significant trade-off. Our key research focuses, their corresponding goals and metrics, as well as some simplified versions of result visualizations are presented in [Figure 1.1](#). More details can be found in [chapter 4](#).

The analysis and results help deepen our understanding of how model robustness is inherited and gained through fine-tuning and offer practical insights for improving model safety while minimizing the impact on standard accuracy.

## 2 BACKGROUND

---

In this section, we describe the architecture of Vision Transformers [3], how parameter-efficient fine-tuning strategies are applied to them, and previous studies on model robustness.

### 2.1 Vision Transformer Architecture

The Vision Transformer (ViT) [3] architecture extends the transformer model [2] from NLP to the image domain in CV. The key component of both architectures is the self-attention mechanism, which captures global dependencies and relationships of the training data to show strong performance across various tasks.

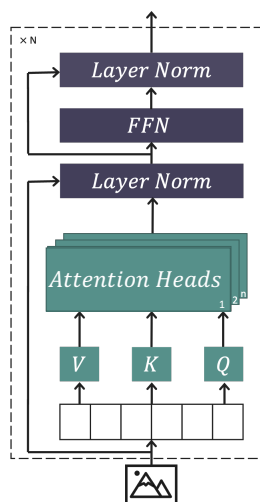
As shown in [Figure 2.1](#), an image is divided into fixed-size patches, each of which is flattened into a vector and projected onto a higher-dimensional space (i.e., embeddings). For each input patch embedding, three matrices – Value (V), Key (K), and Query (Q) – are computed by [Equation 2.1](#),

$$V = XW_V, K = XW_K, Q = XW_Q \quad (2.1)$$

where  $X$  represents the patch embeddings, and  $W_V$ ,  $W_K$ , and  $W_Q$  are learned weight matrices for the V, K, and Q, respectively. The core of the attention mechanism is the calculation of the attention scores (as shown in [Equation 2.2](#)), which measure the relevance of each patch among others.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2.2)$$

The softmax function here converts attention scores into a probability distribution that determines the contribution of each value vector to the output. Multi-head attention extends this idea by performing multiple self-attention operations in parallel. Each attention head processes the input



**Figure 2.1: Vision Transformer Architecture.** This is an architecture visualization of one ViT block.

data differently, enabling the model to focus on different parts of the input simultaneously and enhancing its ability to capture intricate dependencies and patterns. After the attention mechanism, the output is passed through a feedforward neural network (FFN), which includes trainable weights and locates in the middle of two layer normalizations (LN). As is standard practice, an FFN layer consists of linear transformations followed by a Gaussian Error Linear Unit (GELU) activation function, with LN applied to stabilize embedding dynamics and accelerate the convergence of ViT.

The described mechanism is one basic layer of ViT, which stacks multiple such layers together to build increasingly complex representations of the input image. The ViT base model used in our experiments consists of 12 layers, and diverse kinds of parameter-efficient fine-tuning strategies will be applied to different locations (i.e., weights).



Location	Insertion	Reparameterization	Selection
$W_Q, W_K, W_V$	FFN and Activation	matrix multiplication	binary mask
$W_O$	element-wise vectors (parallel)	Kronecker product	Fisher information
multi-head attn	-	Fastfood transform	biases
FFN	-	-	layer selection

**Table 2.1:** Categorization of state-of-the-art PEFT techniques

## 2.2 Parameter-Efficient Fine-Tuning Strategies

Full fine-tuning (i.e., updating all model parameters) and linear probing (i.e., updating only the last classification layer) have been prevalently used [29, 30] to transfer knowledge of pre-trained models to solve downstream tasks in computer vision. While parameter-efficient fine-tuning strategies (PEFTs) have become popular in NLP, a small subset have been adapted to solve CV tasks [31]. The goal of PEFTs is to achieve the same, if not better [32], accuracy as full fine-tuning does but with less training data and fewer trainable parameters for less computational and memory costs. There are different ways to classify PEFTs—by (1) their underlying approach or (2) their primary objectives to minimize memory footprint or only storage [32]. In this thesis, we focus on the categorization based on their underlying mechanism, with which they can be categorized into three groups as presented below. A detailed decomposition based on this categorization is shown in Table 2.1. Given there is no standard categorization rule [32], the PEFTs studied in this thesis and their corresponding categorization are further customized and discussed in Table 3.1.

## Additive Methods

Additive methods focus on augmenting existing pre-trained models with newly-introduced parameters or layers and training only the added parameters. This strategy can largely decrease training time and enhance memory efficiency by reducing the number of gradients to compute for the optimizer during fine-tuning.

There are two major categories: adapter-like methods and "soft prompts". The former [12, 15, 33, 34] inserts standalone modules/adapters, neural networks of negligible size compared to that of the foundation models. The formula is shown in Equation 2.3. For example, Adapters [12] first freeze the original network and then inject the new modules (with specific structures) to be trained on the downstream task, while [34] proposes a mixture-of-experts fashion of leveraging multiple adapter modules. The methods can add the modules to different layers of the pre-trained model. Conventionally, because of the concentration of information of certain layers of ViTs, adapter modules are added sequentially after the attention layer and after the FFN layer, respectively [12].

$$x \leftarrow x + f(xW_{\text{down}}) \cdot W_{\text{up}} \quad (2.3)$$

Here,  $W_{\text{down}}$  and  $W_{\text{up}}$  down-projects and up-projects the original matrix, respectively, and  $f(\cdot)$  is a nonlinear activation function in between the projections. A residual connection is used to wrap up the adapter module.

The second major category is called "soft prompts", which prepend/append parameters to the existing matrices and fine-tune only the added ones. Unlike adapter-like methods, these prepended/appended parameters are not standalone modules but are concatenated with the original weights. They are called "soft prompt" because these strategies are initially designed in NLP by prepending *trainable* parameters to the *prompts* which are then fed to a language model during fine-tuning. The same logic

can be transferred to the image domain. Prefix tuning [8] (denoted in Equation 2.4) prepends tunable parameters to the Key and Value matrices of attention layers, while Prompt tuning [14] only prepends trainable parameters to the input embeddings. The added parameters will be updated directly through gradient descent during fine-tuning.

$$\text{head}_i = \text{Attn}(xW_q^{(i)}, \text{concat}(T_k^{(i)}, W_k^{(i)}), \text{concat}(T_v^{(i)}, W_v^{(i)})) \quad (2.4)$$

Here,  $W_q^{(i)}$ ,  $W_k^{(i)}$ , and  $W_v^{(i)}$  are queries, keys, and values of the  $i$ -th head,  $T_k^{(i)}$  and  $T_v^{(i)}$  are two sets of trainable prefix vectors added to the key and value matrices, and  $x$  is the input of the multi-head attention layer.

## Selective Methods

Selective methods [11, 17, 35] can also be referred to as “sparse update methods”. They ignore the model architecture but select parameters based on either their types or relative importance to various downstream tasks. For example, for each layer of the pre-trained model, BitFit [17] fine-tunes only the bias terms of a model, leaving all other weights unchanged.

## Reparameterization-Based Methods

Reparameterization-based methods leverage intrinsic dimension (i.e., the minimum dimension of model weights required for a model to solve downstream tasks), which is derived from the weights of the pre-trained network, to minimize the number of trainable parameters. Counterintuitively, previous works [7, 10, 36] find that the size of the subspace that needs to be fine-tuned is smaller for bigger pre-trained models. This property makes this type of PEFT strategy more effective with large foundation models. LoRa [7] (as shown in Equation 2.5) decomposes the weight matrices of the pre-trained models into a product of two low-rank (much smaller) matrices, and then adds those two trainable matrices back to the

attention layers to approximate weight updates. Continuing on this line of work, instead of using matrix multiplication, KronA [10] proposes to use the Kronecker product for matrix factorization to further reduce the number of trainable parameters during fine-tuning.

$$x \leftarrow W_0x + \Delta Wx = W_0x + s \cdot W_{\text{down}}W_{\text{up}}x \quad (2.5)$$

Here,  $W_0$  and  $\Delta W$  are the weights of pre-trained model and the weight updates, respectively.  $s \leq 1$  is a tunable scalar hyperparameter, and  $W_{\text{down}}$  and  $W_{\text{up}}$  are trainable parameters for matrix reparameterization.

## 2.3 Model Robustness

In this thesis, we study the robustness of models fine-tuned by PEFT methods from both security and safety perspectives. Adversarial robustness [28, 37, 38], as one of the major subareas of machine learning security, focuses on measuring and enhancing model resilience against adversarial examples—inputs to models with minute, often imperceptible-to-humans perturbations that trigger model misclassification. We focus on white-box evasion attacks and choose PGD, a representative attack algorithm widely used in security evaluations, for our experiments. In terms of safety [1, 39, 40], we study models’ natural vulnerabilities, their fidelity facing domain shifts. Those out-of-distribution shifts are inevitable in realistic settings where models are deployed, and, thus, it is crucial to take them into consideration for a more comprehensive evaluation of the models. Here, we briefly discuss the prevalent adversarial and natural attacks in the field that we integrated into our evaluation.

## Adversarial Attacks

Instead of updating model weights by minimizing pre-defined loss through backpropagation during training, white-box evasion attacks freeze model weights and use similar optimization techniques to update the *input* with different objectives—either *maximize* model loss [28, 37, 41, 42] or *minimize* perturbation require for misclassification [38]. Basic Iterative Method (BIM) [41] extends the same heuristic of fast gradient sign method (FGSM) [37]. It is an iterative process that follows gradual perturbation steps written as:

$$x_{t+1} = \Pi_{x+\mathcal{B}}(x_t + \alpha \text{sgn}(\nabla_x L(\theta, x, y))) \quad (2.6)$$

where  $L$  is the loss function associated with the model and its parameters  $\theta$  being attacked, original input  $x$ , and label  $y$ . Using gradient ascent with step size  $\alpha$ , BIM iteratively generates perturbations while staying within a predefined perturbation budget  $\mathcal{B}$  outlined by an  $\ell_\infty$ -norm. While project gradient descent (PGD) [28] also uses the iterative method, it applies *Random Restart*, wherein inputs are initially randomly perturbed within an  $\ell_\infty$  ball. We refer to the original attack papers for more details.

## Out-of-Distribution Shifts

Domain shifts here focus on the shifts of the style and background from training images to test images, designed to test if the model learns key features for classes. For example, as a standard domain adaptation dataset, DomainNet [1] transfers from sketch images to real, clip art, and painting images of the same class. CIFAR-10 [43] is used as the training dataset while STL [40] and CIFAR-10.1 [44], which shares a similar data collection protocol but exhibits a minute distributional shift, are used as test datasets. Due to the large computational requirement for analyzing the robustness-accuracy trade-off space throughout the fine-tuning process, we focus on using DomainNet for our OOD evaluation.

### 3 METHODOLOGY

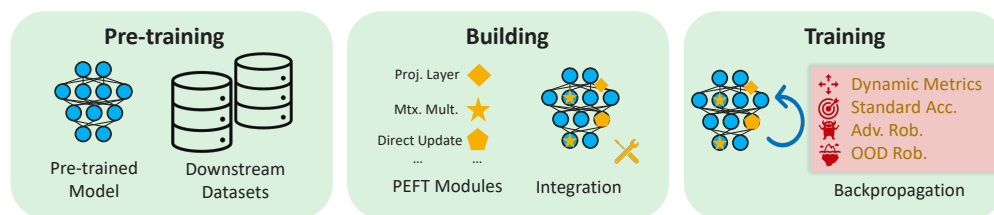
---

In this section, we outline the considered threat model in security and safety settings. While PEFTs are designed to extract and fine-tune certain knowledge to optimize a model’s standard accuracy on downstream tasks, we aim to study how updating the knowledge impacts model robustness. We select representative PEFT strategies in order to characterize the parameter space of PEFTs. Furthermore, we explain the rationale of our designed sensitivity analysis with the presented framework (Figure 3.1) to study how sensitive the robustness-accuracy trade-off (with both adversarial and OOD data) is with respect to fine-tuning strategies, the number of backpropagations, and downstream tasks.

#### 3.1 Threat Model

We consider both security and safety perspectives to evaluate model robustness with different fine-tuning approaches. A key distinction between these settings is that, in security, adversaries actively attempt to exploit model weaknesses, whereas for safety, no attacker is present—robustness is, instead, tested against distributional shifts that occur in realistic environments to measure models’ generalization capability.

For adversarial robustness, we consider the worst-case adversary through a “white-box” threat model, where adversaries have full access to the model architecture and parameters to generate perturbations. Specifically, these perturbations are produced by iterative attack algorithms under some  $\ell_p$ -norm, which limits the magnitude of the perturbation, while sufficient to induce model misclassifications, are imperceptible to the human eye. We consider  $\ell_\infty$ -norm in the thesis. For safety robustness, no targeted perturbations are applied, and test images are sourced from different domains (e.g., sketches vs. real images) that are different from

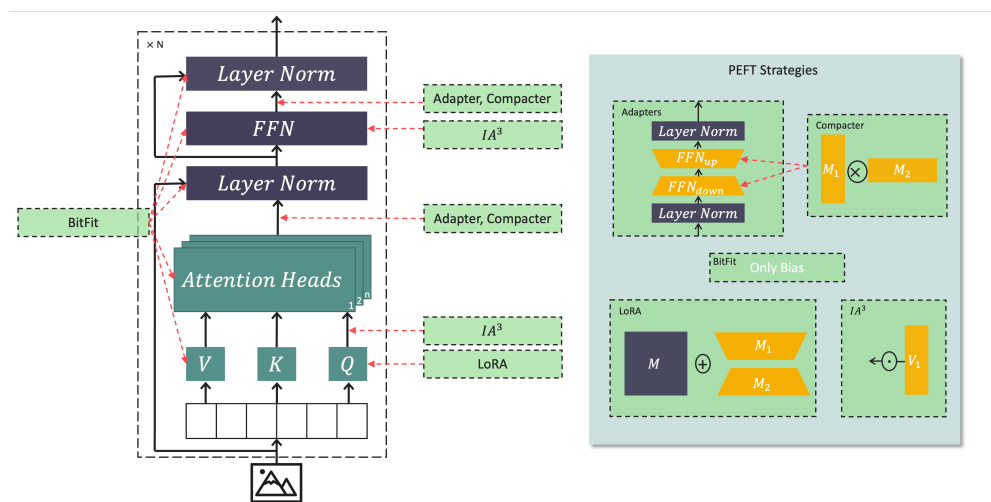


**Figure 3.1: A systematic framework for sensitivity analysis of PEFTs’ robustness and accuracy.** A framework to characterize the trade-off space of PEFT strategies. (1) Select pre-trained models and downstream tasks; (2) Integrate PEFT modules to the pre-trained model; and (3) Dynamically evaluate their robustness and accuracy across backpropagations.

the training data, simulating practical distributional shifts.

## 3.2 The Space of PEFTs

One key challenge in evaluating the robustness of models fine-tuned with various PEFT strategies is the systematic characterization of the parameter space in which they reside. The location of a PEFT strategy in the parameter space determines the knowledge a model eventually obtains, which further determines its capability—robustness and accuracy—on downstream tasks. We probe the space by selecting five SOTA PEFTs from the previously-mentioned categories (i.e., BitFit from selection, LoRA and (IA)<sup>3</sup> from reparameterization, and Adapter and Compater from insertion). In addition, we study the space by analyzing these strategies through two key dimensions: the *information* in the pre-trained models they extract and the *mechanisms* they use to fine-tune the extracted information. In [Figure 3.2](#), we draw out where PEFT strategies are applied to a ViT block (left) and the underlying mechanisms they use (right). Note that the extracted information location and mechanisms remain the same throughout N ViT blocks. More formally, we derive a table of the PEFT parameter space we considered ([Table 3.1](#)).



**Figure 3.2:** Left: A graphical illustration of how five different PEFTs are applied to one ViT block. Right: Individual PEFT mechanism.

## Extracted Information

We study the extracted information, which is contained in model weights and intermediate representations, based on both its type and location. The primary difference between parameter-efficient and traditional fine-tuning strategies is that, instead of fine-tuning all pre-trained model weights, PEFTs strategically extract pre-trained information by inserting/selecting a small amount of parameters to be fine-tuned to optimize model’s standard accuracy. However, what type of information is chosen and where to choose it vary across PEFTs. By characterizing and probing these two directions, we study if fine-tuning different information extracted from pre-trained models could impact model robustness on downstream tasks.

With the selected PEFTs, we observe that there are two types of information extracted—model weights and input representations. Model weights are the parameters of the pre-trained model layers (i.e., static), while input representations are the output of the intermediate layers after feeding input to the model (i.e., dynamic, depending on the input at run-



time). It is straightforward for certain strategies such as LoRA [7], which operates directly on the attention matrices by decomposing them, while the line is harder to draw for others such as (IA)<sup>3</sup> [15], which rescales the weights (i.e., attention weights and FFN activations) with inserted vectors, but it is equivalent to rescaling the representations.

Furthermore, both the weights and the representations can be obtained from different parts of the model (e.g., attention weights and biases). Among them, attention weights, which the ViT relies on entirely to draw global dependencies between input and output [2], have been studied the most in transfer learning for simplicity and parameter-efficiency [7, 12, 45]. We characterize the information extracted based on the commonly chosen model layers (i.e., attention weights, feed-forward neural layers, and biases) and the corresponding input representations obtained after those layers in the first five columns of Table 3.1.

The Space of PEFT									
PEFT Strategies	Information Location					Mechanism			
	Attn	Rep.	FFN	Rep.	Bias	Proj. Layers	Matrix Reparam	Element-wise Mult.	Direct Update
LoRA	●	○	○	○	○	○	●	○	○
IA3	○	●	○	●	○	○	○	●	○
Adapter	○	●	○	●	○	●	○	○	○
Compacter	○	●	○	●	○	●	●	○	○
BitFit	●	○	●	○	●	○	○	○	●

**Table 3.1:** The space of PEFT strategies in terms of information location and underlying mechanisms

## Underlying Mechanisms

After the information is extracted from the pre-trained model, different mechanisms of PEFTs are designed to update the corresponding parameters to adapt the model to downstream tasks. We find three main approaches—1) projection with neural layers: feedforward neural layers

and layer normalizations are used to down- and up-project the intermediate representations; 2) matrix/vector computation: specifically, multiplication is used here to rescale the matrices to which it is applied; 3) direct update: backpropagation is directly applied to update the information extracted without additional techniques.

We find that these mechanisms are partially related to the category (i.e., selection, reparameterization, and insertion) of the corresponding PEFTs. Projection layers are usually *inserted* to the pre-trained model, while matrix multiplication are used to *reparameterize* the original large matrix to multiple smaller matrices, and *selected* parameters are usually directly updated. However, it is worth noting that the mapping is not definite, one PEFT strategy can combine different mechanisms. For instance, Compacter [45], an insertion-based method, uses matrix multiplication to decompose inserted matrices instead of the original weights.

### 3.3 Sensitivity Analysis for Robustness

#### Motivation

As shown in [Figure 3.1](#), we propose a framework for systematically analyzing the sensitivity of robustness of PEFT strategies to key factors, including the parameter space of PEFTs, the number of training updates, and different downstream tasks. Despite the rapid emergence of new PEFT methods and associated libraries, no existing system examines how the relationship between robustness and accuracy evolves during fine-tuning. Our framework addresses this gap by offering a structured approach to study it dynamically throughout the fine-tuning process.

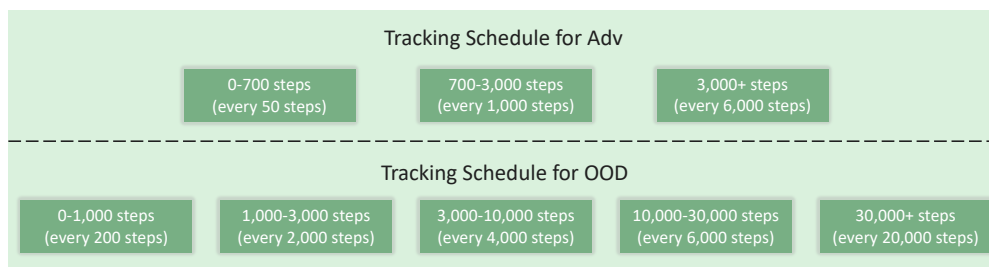
Our study shares similarity to previous work on classic phenomenon of overfitting. Overfitting, as described by [46], involves a divergence between training and validation/test accuracy: validation accuracy declines as training accuracy continues to improve, reflecting the model’s

over-specialization on the training data, including noise and random fluctuations, rather than generalizable patterns. Extensive research has explored this issue, leading to the development of early stopping methods [47, 48]. Those methods serve as implicit regularizers, improving test accuracy and reducing computational costs. Similarly, during fine-tuning, pre-trained models may learn *non-robust* features from downstream datasets, potentially compromising robustness while training accuracy improves.

However, the relationship between this robustness-accuracy trade-off and the well established trade-off between test vs. training accuracy remains unclear. Note that the robustness-accuracy trade-off in the PEFT space is also different from the traditional training schemes such as training from scratch—while the pre-trained models already have general knowledge on certain downstream concepts, it is not aware of the non-robust features, which are specific patterns/distributions of downstream datasets as opposed to the pre-training datasets. Furthermore, it is important to study how many robustness properties of the few selected features, optimized for standard accuracy on downstream datasets, can be inherited or gained during fine-tuning. Such insights can lead to more robustness-aware and efficient fine-tuning approaches.

## Tracking Schedule

Measurement frequency and analysis metrics are central to our framework to effectively capture the robustness-accuracy trade-off space and compare across PEFT strategies. We initially select training epochs as the standard monitoring unit to dynamically track model robustness and accuracy. However, for downstream tasks closely aligned with upstream datasets, models can often achieve 80% of their converged accuracy and robustness within the first few epochs. This rapid convergence makes it challenging to capture nuanced dynamics during the early stages of training, potentially missing key behaviors due to the sparsity of measurements.



**Figure 3.3: Tracking schedule for standard accuracy and Adv and OOD robustness.** Designed tracking schedule for efficiently attacking/evaluating model states during fine-tuning.

$$\# \text{ backpropagation steps} = \frac{\text{size of the training dataset}}{\text{batch size}} \quad (3.1)$$

To address this, we propose focusing on the progress at the granularity of individual updates rather than full epochs, which aggregate too much information for certain transfer tasks. While the number of epochs determines how many rounds to go through the entire training dataset, the number of backpropagation steps of each epoch (Equation 3.1) is determined by batch size (i.e., how many images to learn from for one backpropagation) and the size of training datasets. However, to track every backpropagation step can be both computationally expensive and/or memory-intensive. For adversarial robustness, perturbations are generated for all test images at each tracking step, incurring high computational costs with attack algorithms. In contrast, tracking out-of-distribution robustness involves constantly loading and feeding all test data outside the training domain into the model, which is memory- and storage-intensive.

Based on trials with different selected tracking intervals, we designed an efficient tracking schedule for model states during fine-tuning. It is tailored to different types of robustness measurement (shown in Figure 3.3). This schedule balances computing feasibility with our research need. By strategically monitoring changes at pre-defined backpropagation steps,

we achieve finer-grained insights into model behavior.

## Pareto Front Curves & AUC

We use Pareto front curves to characterize the robustness-accuracy trade-off space of different PEFT strategies across backpropagation steps. In addition, we use area under the curve (AUC) as a metric to quantitatively compare these trade-off spaces. Pareto-based approaches have been widely adopted in machine learning to study trade-offs, especially robustness-accuracy trade-offs, in various contexts [49, 50, 51]. Pareto front curves are particularly effective because they visually represent the set of optimal solutions where no objective, such as robustness or accuracy, can be improved without sacrificing the other. In the context of robustness-accuracy trade-offs, this method enables clear identification of how different strategies perform relative to one another by highlighting strategies that balance the competing objectives most effectively.

AUC [49, 51] complements this analysis by offering a single scalar metric to summarize the trade-off space captured by the Pareto front. Larger AUC values indicate a better overall trade-off, as the curve spans a broader range of robustness-accuracy pairs. Together, Pareto front curves and AUC allow for both detailed visualization and concise summarization of the trade-off space across PEFT strategies.

We employ these two methods to get the optimal robustness-accuracy Pareto front curves for each PEFT strategy for various downstream datasets. This will examine whether the hypothesized trade-off phenomenon exists as models are fine-tuned for different downstream tasks. Further, we derive their corresponding AUC to compare the trade-off space across strategies in order to investigate whether and why different strategy offer different trade-off space.

## 4 EVALUATION

---

With our framework of model pre-training, building, and fine-tuning with dynamic sensitivity analysis, we ask three major research questions:

- RQ1** Is model accuracy still at odds (as it is in traditional training [52]) with robustness against adversarial examples in the paradigm of pre-training and parameter-efficient fine-tuning?—*How much* should we fine-tune to get a both accurate and robust model?
- RQ2** Do different PEFTs offer different *combined robustness and accuracy* while all aims to optimize standard accuracy on downstream tasks?
- RQ3** Are findings invariant to *safety* OOD contexts?

We fine-tuned 231 models across 6 datasets (i.e., 5 adversarial robustness benchmark datasets and 6 out-of-distribution benchmark domains) with 7 fine-tuning strategies (i.e., 5 parameter-efficient fine-tuning strategies and 2 traditional baseline fine-tuning methods) for 3 runs. While training each model, we evaluate the intermediate model state on average 20 times for adversarial robustness (around 2,100 PGD attacks) and 16 times for OOD robustness (around 2,016 OOD evaluations).

### 4.1 Experimental Setup

We perform our experiments on twelve NVIDIA A100 GPUs with 40 GB of memory, with the support of other available GPUs from the Center of High Throughput Computing [53]. We describe the pre-trained model, PEFT strategies, datasets, and security and safety measures used below.

**Table 4.1:** Configurations of PEFTs based on standard practices

PEFTs	Configs	Values
Adapter & Compacter	reduction factor	8
	non linearity	gelu
	locations	multi-heads attn, $W_O$
(IA) <sup>3</sup>	locations	$W_K, W_V, \text{FFN}$
LoRA	locations	$W_K, W_V, W_Q, W_O$

## Pre-trained Models

We use ViT-Base model, pretrained on ImageNet-21k [54] (14 million images, 21,843 classes) at resolution 224x224 from HuggingFace [25].

## PEFTs and Training Configurations

We select five widely-used parameter-efficient fine-tuning strategies—Adapter [12], Compacter [45], BitFit [17], LoRA [7], and (IA)<sup>3</sup> [15]—from the three categories to probe the parameter space of PEFT, as well as the two traditional strategies—full fine-tuning and linear probing—as our baselines. We use AdapterHub library [55] for integrating various PEFT modules to the pre-trained model architecture. Those module configurations are adjusted based on common practice in solving CV tasks [31, 56], while certain parameterization is explored as described in Table 4.1. Grid search is used to find training configurations (i.e., base learning rate -  $\{1e-4, 1e-5, 3e-5, 5e-5\}$  and base weight decay -  $\{1-e2, 1-e3\}$ , together with adjustment ratios for each PEFT strategy  $\{1, 10, 5, 10, 2, 2, 3\}$  (as the order of PEFTs shown in Table 4.2) based on previous literature [7, 12, 17] for PEFTs and downstream tasks. Due to the size of the OOD dataset, we set the base weight decay to be  $1e-2$  for efficiency. The detailed information of PEFTs configurations and optimal training hyperparameters based on grid search can be found in Table 4.1, Table 4.2, and Table 4.3.

**Table 4.2:** Strategy configurations with datasets (Adv)

Fine-Tuning Methods	Learning Rate / Weight Decay for Adv Exps.				
	CIFAR10	CIFAR100	CUB200	Caltech256	Stanford Dogs
Full Fine-tune	3e-5/1e-3	5e-5/1e-3	5e-5/1e-3	1e-4/1e-3	1e-4/1e-3
Linear Probe	1e-5/1e-3	1e-5/1e-2	5e-6/1e-3	1e-5/1e-3	1e-5/1e-3
LoRA	5e-4/1e-2	5e-4/1e-2	2.5e-4/1e-2	5e-4/1e-3	5e-5/1e-2
BitFit	1e-5/1e-4	1e-5/1e-3	5e-6/1e-4	1e-5/1e-3	1e-5/1e-3
Adapter	1e-4/1e-3	2e-4/1e-2	2e-5/1e-2	2e-4/1e-2	2e-4/1e-3
Compacter	2e-4/1e-3	2e-4/1e-3	1e-4/1e-3	2e-4/1e-3	2e-4/1e-3
(IA) <sup>3</sup>	1.5e-4/1e-3	3e-4/1e-2	3e-4/1e-3	3e-4/1e-3	1.5e-4/1e-3

**Table 4.3:** Strategy configurations with datasets (OOD)

Fine-Tuning Methods	Learning Rate for OOD Exps.					
	Clipart	Infograph	Painting	Quickdraw	Real	Sketch
Full Fine-tune	1e-4	1e-4	1e-4	1e-4	1e-4	1e-4
Linear Probe	1e-3	1e-3	1e-3	1e-3	5e-4	1e-3
LoRA	5e-4	2.5e-4	5e-4	2.5e-4	5e-4	5e-4
BitFit	1e-3	1e-3	5e-4	1e-3	5e-4	1e-3
Adapter	2e-4	2e-4	2e-4	1e-4	2e-4	2e-4
Compacter	2e-4	2e-4	2e-4	2e-4	2e-4	2e-4
(IA) <sup>3</sup>	3e-4	3e-4	3e-4	3e-4	3e-4	3e-4

## Benchmark Datasets

We use six different datasets in our experiments in order to cover (1) a wide range of image categories—both coarse classes (e.g., birds, dogs, cars, etc.) and specific species within one class (e.g., 200 species of birds), (2) a common attack algorithm, PGD, for adversarial robustness evaluation, and (3) a standard safety evaluation benchmark. We provide a summary of these datasets in [Table 4.4](#) and detailed descriptions below.

### Adversarial Robustness

**CIFAR10.** CIFAR10 [44] is a dataset for image classification. It has been extensively used as a benchmark in security and safety machine learning literature. CIFAR10 consists of 10 classes, with 6,000 images per class (5,000 training and 1,000 test images).

**CIFAR100.** CIFAR100 [57], similar to CIFAR10, has 100 classes, with 600 images per class (500 training and 100 test images). Those 100 classes



Fine-tuning datasets	Robustness	
	Adversarial ( $\alpha = 0.25, \epsilon = 1, \text{step}=15$ )	Out-of-distribution
CIFAR10	PGD applied on test set	-
CIFAR100		-
Caltech256		-
CUB-200-2011		-
Stanford Dogs		-
DomainNet		-

**Table 4.4:** Datasets and robustness measures

have their own "fine" label as their class and are also grouped into 20 superclasses as their "coarse" label.

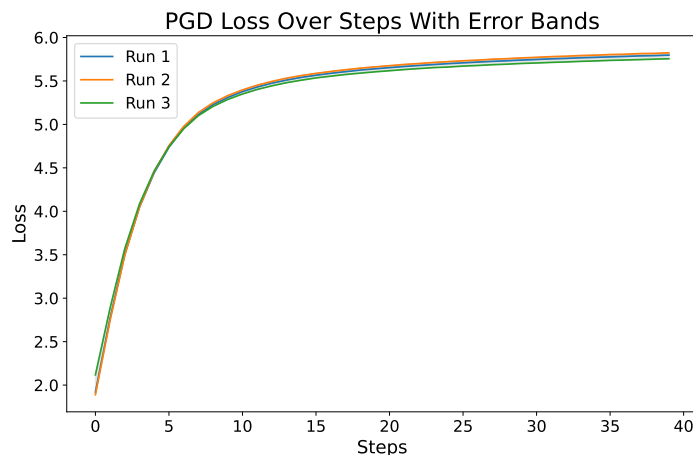
**Caltech256.** Caltech256 [58] contains 256 classes with a total number of 30,607 images. It claims to have improved image qualities for machine learning tasks by, for example, avoiding image rotation and increasing the minimum number of images in any class from 31 in Caltech101 [59] to 80. We only use RGB images from the dataset while getting rid of grayscale images for our experiments because the processor of the pre-trained model expects input images to have three channels.

**CUB-200-2011.** CUB [60] contains 11,788 images of 200 bird species. Different from other datasets, it contains images from only one of the coarse classes of ImageNet-21k. As a standard adversarial machine learning benchmark, it is also helpful to test the cases where the model is pre-trained on general knowledge but fine-tuned on more fine-grained information.

**Stanford Dogs.** Stanforddogs [61], similar to CUB, is another fine-grained categorization classification problem for machine learning models. It contains 22,000 images of 120 breeds of dogs. As a standard benchmark, it is also used to measure models' adversarial robustness.

### OOD Robustness

**DomainNet.** DomainNet [1] is a standard domain adaptation dataset. It includes six different domains, including clipart, infograph, painting, quick-



**Figure 4.1: PGD loss convergence plot during attacking.** It is one of the attack loss curves on CUB200 dataset with full fine-tuning in order to verify the loss is converging.

draw, real, and sketch. We fine-tune the pre-trained model on a single domain and test it on other domains to get the model’s fidelity on different distributions of images.

### Adversarial Attack Algorithms.

While OOD robustness is measured directly with images from domains which are different from the training domain, robustness against adversarial attacks is measured by attacking a model with generated adversarial examples using a selected attack algorithm. We use a widely-regarded state-of-the-art white-box evasion attack algorithm PGD to craft adversarial examples with the test datasets of the downstream tasks. Details of the attack algorithms and variables are described in [Section 2.3](#).

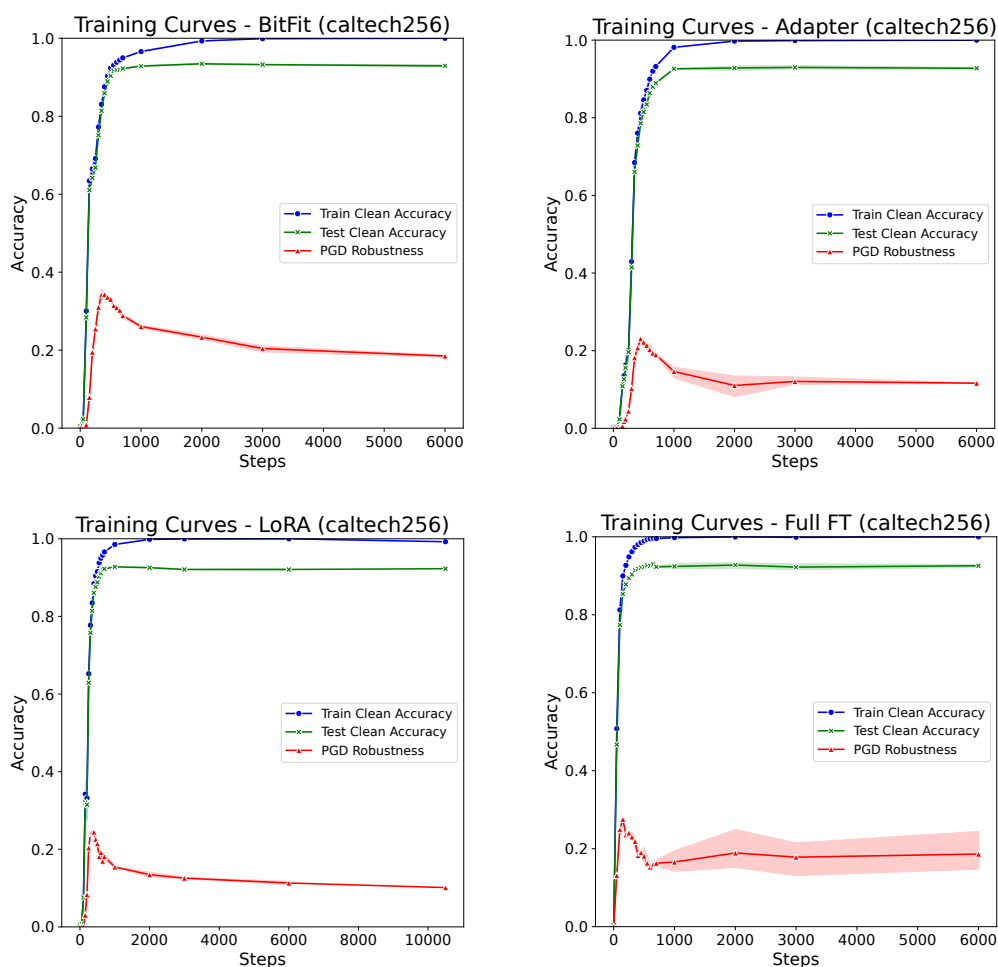
PGD uses a clipping method with a hyperparameter  $\epsilon$  to restrict the amount of perturbation  $\delta$  added. We explore a range of budget values  $\epsilon$  (i.e., 1, 2, ..., 8/255) and study the percentages of the budget that PGD actually consumed  $\delta/\epsilon$ . We choose  $\epsilon = 1/255$  [62], step size  $\alpha = 0.25/255$ , and

the number of steps to be 15. We verify that the attack loss is converging for around 95.6% at step 15 as shown in [Figure 4.1](#).

## 4.2 Trade-off Between Accuracy and Robustness

Our study on the trade-off between standard accuracy and robustness within the paradigm of pre-training and parameter-efficient fine-tuning is motivated by prior research on similar phenomena in traditional training settings (i.e., training models from scratch). Those studies have claimed that robustness is at odds with accuracy both empirically [52, 63] and theoretically [52, 64, 65]. Previous work [22] attributes adversarial examples to the presence of "non-robust" features—patterns that, while highly predictive for standard accuracy, are imperceptible to humans—which models extract from training data distribution. Given the fundamental assumption in computer vision that training and test data are drawn from the same distribution [66, 67], this reliance on non-robust features explains why model robustness on test data often degrades as models optimize for standard accuracy on the corresponding training data.

However, with the advent of transfer learning and the adoption of PEFT strategies, the fundamental assumption of prior studies no longer holds. In this paradigm, upstream pre-training datasets often differ significantly (e.g., classes/labels, domains, sizes, etc.) from the downstream data on which models are fine-tuned and evaluated. This shift necessitates a reassessment of the trade-off between accuracy and robustness. We hypothesize that the trade-off phenomena still exist but exhibit differently: pre-trained models initially are more resistant to adversarial examples generated on downstream datasets. As fine-tuning progresses and models are adapted more to downstream data, their robustness peaks and subsequently declines. Since PEFTs only update a small portion of the pre-trained model, how much robustness can be inherited and/or gained



**Figure 4.2: The trend of training standard accuracy (blue), test standard accuracy (green), and adversarial robustness on test dataset (red) across the number of backpropagation steps on Caltech256. PGD robustness reaches its peak and drops at an early stage of training, while both training and test clean accuracy keep increasing and plateau in the end.**

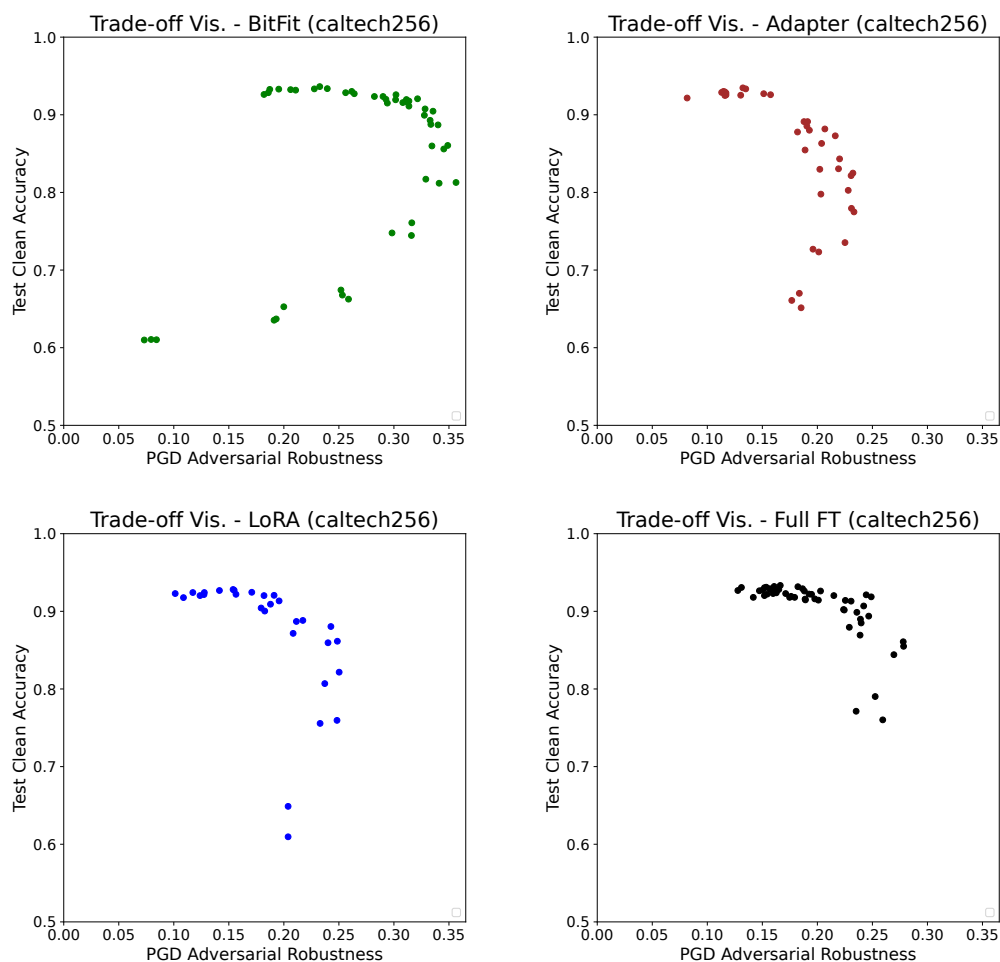
and how much non-robust features are exploited for standard accuracy throughout training are important to be investigated for better practice. Our experiments described below are designed to answer those questions.

## Experimental Results

We observe a consistent robustness-accuracy trade-off emerging early during fine-tuning across all experimental settings, which include 5 PEFT strategies, 2 baseline methods, and 5 datasets. We conduct a dynamic sensitivity analysis with the designed tracking schedule (Figure 3.3) integrated into our experimental pipeline (Figure 3.1). For example, Figure 4.2 shows results on Caltech256 with BitFit, Adapter, LoRA, and full fine-tuning (complete results in Section 8.1). During fine-tuning, training and test accuracy increase exponentially from  $\sim 0\%$  to  $\sim 100\%$  and  $\sim 90\%$ , respectively, converging within 1,000 steps. In contrast, PGD robustness exhibits a peak of 25% around step 400 before declining to  $\sim 10\%$  at convergence.

The turning points on the robustness curves prove the existence of the trade-off and strongly suggests that models begin to learn non-robust features early in the fine-tuning process. While these features improve standard accuracy, they also introduce vulnerabilities that reduce robustness. The initial increase in robustness before these turning points, coupled with the steep rise in standard accuracy, indicates that PEFT efficiently adapts pre-trained knowledge and newly introduced parameters to downstream datasets. This process effectively transfers both standard knowledge and robustness from the pre-trained models. Notably, the corresponding high standard accuracy observed at the peak of model robustness suggests a promising direction: accelerating the learning of robust and useful features before the peak while constraining or regularizing the emergence of non-robust yet predictive features afterward could preserve more robustness with minimal impact on standard accuracy.

Furthermore, we analyze the robustness-accuracy trade-off from the training curves by focusing on the behavior around the turning point, restricting the standard accuracy to above 50%. As shown in Figure 4.3, the scatter plots for Caltech256 illustrates a distinct trade-off with each dot corresponding to a specific timestamp (i.e., backpropagation step) for



**Figure 4.3: Trade-off visualization between standard accuracy and robustness for Caltech256.** The dots are corresponding to different time stamps during training (from bottom left to upper right to upper left as time goes on).

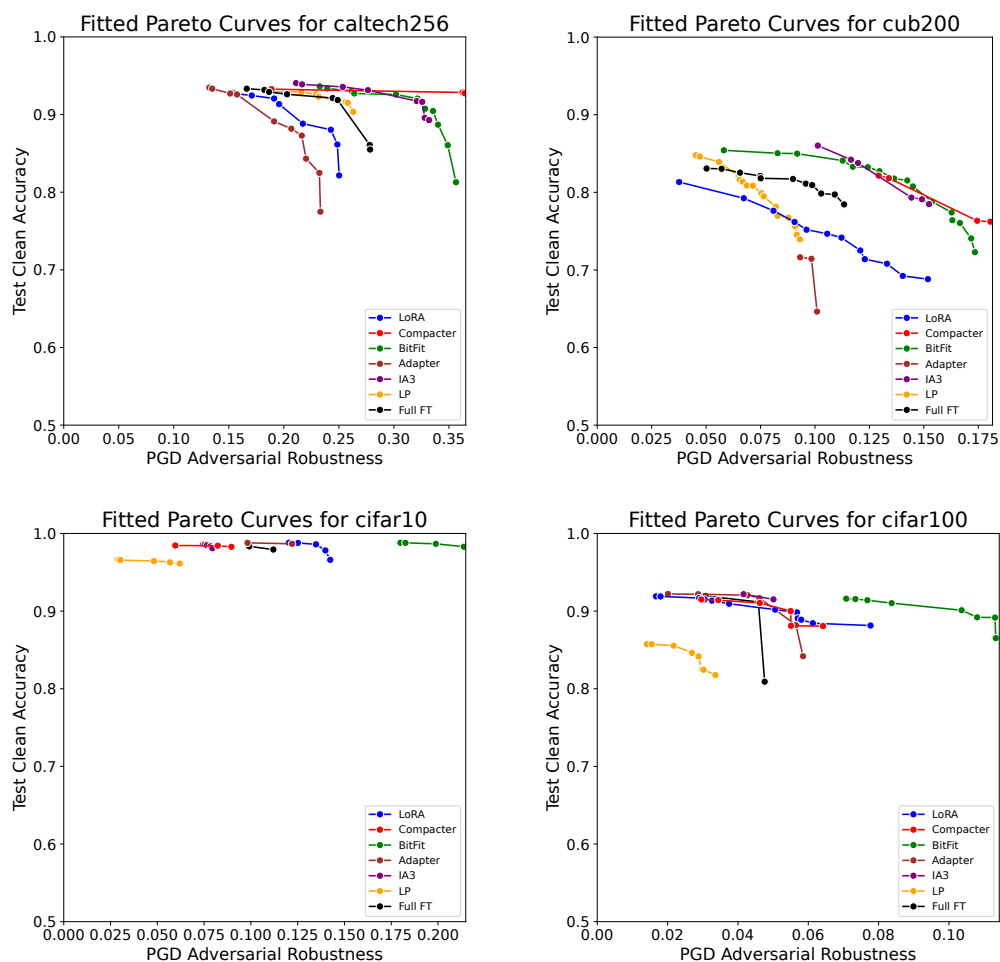
three runs). Early in training, the models show low accuracy and robustness, with points clustered in the bottom-left corner. Over time, the points move upward and to the right, indicating simultaneous improvements in standard accuracy and robustness. However, as training progresses, robustness peaks and then declines, while accuracy continues to increase,

with points shifting toward the top-left corner. This trend is particularly evident in BitFit and full fine-tuning, where standard accuracy exceeds 90% at convergence, but robustness drops significantly—approximately 15% (from 35% to 20% and from 25% to 10%, respectively). In comparison, other methods exhibit smaller declines in robustness when standard accuracy surpasses 90%. These findings highlight that PEFT strategies targeting information in or around attention layers achieve a better balance between robustness and accuracy compared to methods that fine-tune all weights or only the biases, while the latter methods achieve higher robustness at its peak.

### 4.3 Pareto Front Curves in the Trade-off Space

After analyzing the consistent robustness-accuracy trade-off phenomenon across our experimental settings, we further ask how sensitive these robustness-accuracy trade-offs are to different downstream tasks and fine-tuning strategies while focusing only on the Pareto front curves in the trade-off space. In terms of downstream task complexity, we consider both the number of classes and features and the similarity between classes of downstream tasks and those of the upstream datasets.

First, we derive the Pareto curves from [Figure 4.3](#) by identifying the set of points that represent optimal trade-offs between standard accuracy (y-axis) and PGD adversarial robustness (x-axis)—a point is considered optimal if no other point has higher accuracy and robustness simultaneously. This approach effectively captures the trade-off frontier for each fine-tuning strategy. After constructing the Pareto front curves for each method, we aggregate them for each dataset. The resulting curves for Caltech256, CUB200, CIFAR10, and CIFAR100 are shown in [Figure 4.4](#). Note that the ranges of robustness vary across downstream tasks. In order to focus on comparing Pareto front curves across different PEFT strategies,



**Figure 4.4:** The Pareto front curves of the trade-off between standard accuracy and robustness on Caltech256, CUB200, CIFAR10, and CIFAR100. The Pareto front curves of different PEFT strategies reside in different locations in the trade-off space.

we set different x-axis values for each downstream dataset.

The trends observed in the plots reveal significant differences across downstream datasets. For Caltech256 and CUB200, the trade-offs are more pronounced, with robustness peaking before sharply declining as standard accuracy approaches the final 10% of its convergence. In comparison,



CIFAR100 exhibits a more gradual trade-off, characterized by smaller gradients (in absolute values). CIFAR10 demonstrates the smallest and most gradual trade-off, with robustness remaining relatively stable as accuracy improves by the last 2%. We notice that the 10 classes of CIFAR10 are a subset of ImageNet21k classes [54], while CUB200 benchmark requires a model to distinguish 200 bird species, all within a single high-level class ("Bird") in the ImageNet21k hierarchy. This suggests that more complex and detailed tasks require models to learn intricate features that may introduce vulnerabilities and not align with adversarial robustness. The trend here highlights that the robustness-accuracy trade-offs strongly correlate with the complexity of the downstream tasks and their resemblance to the upstream pre-training dataset.

	cifar100	cifar10	stanforddogs	caltech256	cub200
BitFit	<b>0.1033</b>	<b>0.2112</b>	0.0762	0.3311	0.1447
Adapter	0.0536	0.1205	0.0510	0.2141	0.0722
LoRA	0.0703	0.1406	0.0633	0.2299	0.1168
Compacter	0.0583	0.0882	<b>0.0891</b>	<b>0.3394</b>	<b>0.1467</b>
IA3	0.0462	0.0783	0.0540	0.3108	0.1292
LP	0.0286	0.0599	0.0166	0.2441	0.0769
Full FT	0.0436	0.1103	0.0542	0.2579	0.0933

**Table 4.5:** Area under the curve (AUC) of the Pareto front curves

Furthermore, we examine the sensitivity of different parameter-efficient fine-tuning (PEFT) strategies by computing the area under the Pareto front curves (AUC), as summarized in Table 4.5. BitFit consistently achieves the highest AUC for CIFAR10 ( $\sim 0.21$ ) and CIFAR100 ( $\sim 0.10$ ), indicating superior robustness-accuracy trade-offs for these datasets. For Stanford Dogs, Caltech256, and CUB200, Compacter outperforms other methods with AUC values of 0.0891, 0.3394, and 0.1467, respectively. These results suggest that BitFit excels in less complex datasets, whereas Compacter is more suited to more complex datasets, such as Caltech256, which require more nuanced feature extraction. BitFit’s strong performance on CIFAR

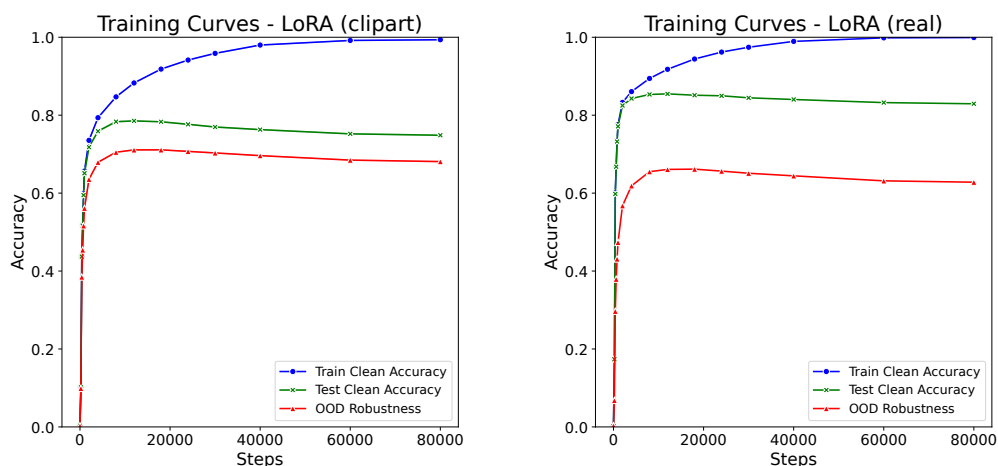
datasets can be attributed to their resemblance to ImageNet21k, where adapting only the bias terms effectively inherit pre-trained features and robustness. In contrast, datasets such as Stanford Dogs, Caltech256, and CUB200 require more detailed feature adaptation. Here, Compacter’s inserted modules with low-rank reparameterization, which focus on intermediate representations after both attention layers and FFNs, enables effective knowledge transfer from upstream pretraining while simultaneously adapting to downstream tasks. This allows Compacter to achieve a better balance between robustness and accuracy.

Notably, linear probe and full fine-tune underperform across nearly all datasets, with the lowest AUC values for CIFAR100 (0.0286 and 0.0436) and CUB200 (0.0769 and 0.0933). This indicates that fine-tuning all parameters or freezing almost all layers introduces instability in managing the robustness-accuracy trade-off, especially for more challenging or detailed datasets like CUB200.

This analysis supports our hypothesis that robustness-accuracy trade-offs are sensitive to both downstream task complexity and fine-tuning strategies. It underscores the importance of aligning different PEFT strategies (i.e., information extracted and their underlying mechanisms) with the complexity and characteristics of downstream tasks to achieve optimal balance between robustness and accuracy.

## 4.4 On Out-of-Distribution Robustness

In this section, we investigate RQ3—whether the findings from studying adversarial robustness generalize to model safety in real-world out-of-distribution (OOD) scenarios. For OOD robustness, the "non-robust" features exploited by adversarial attack algorithms, which attribute to the robustness-accuracy trade-off phenomena in adversarial settings, are absent. Instead, OOD robustness depends on a model’s ability to generalize



**Figure 4.5:** The trend of training standard accuracy (blue), test standard accuracy in the training domain (green), and OOD robustness in other domains (red) across the number of backpropagation steps. The results of two training domains—clip art and real images—from DomainNet [1] are shown here.

beyond the training distribution, which we hypothesize may lead to different behaviors in the pre-training and fine-tuning paradigm compared to adversarial robustness. To analyze this, we track OOD robustness and standard accuracy throughout training, as showing in Figure 4.5 (complete results can be found in Section 8.2).

Unlike the adversarial robustness trends in Figure 4.2, we observe no substantial decline in OOD robustness after it peaks. Both training and in-domain test standard accuracy (blue and green curves) improve steadily, and once converged, remain stable. However, OOD robustness (red curves) initially improves with training steps but plateaus at lower values compared to standard accuracy. Note that both in-domain test standard accuracy and OOD robustness decline slightly after convergence. This, as opposed to a big decrease on robustness while the standard accuracy still increases, can be explained by the traditional overfitting to the



**Figure 4.6: A heatmap of the highest OOD robustness during training across 6 domains.**

training dataset. For the "real" training domain (i.e., which is the closest to the upstream pre-training dataset distributions), OOD robustness converges significantly below in-domain standard accuracy, highlighting the challenge of generalization to other domains.

This behavior contrasts with adversarial robustness, where training on the training data after a certain point often causes robustness to deteriorate after reaching its maximum. We explain this phenomenon by that OOD robustness relies more on learning *transferable features* applicable across distributions, which are less sensitive to overfitting. In adversarial settings, however, robustness is reduced by some specific *low-level non-robust features* from training distributions that may be beneficial to accuracy optimization.

Furthermore, we examine whether OOD robustness is sensitive to dif-

ferent PEFT strategies or training domains. As shown in [Figure 4.6](#), the domains on the y-axis of the heatmap are the domains that the models are fine-tuned on with corresponding PEFT strategies or traditional fine-tuning methods (listed on the x-axis of the heatmap). The values indicate the highest OOD robustness on images outside of the fine-tuning domain models achieved during fine-tuning. Note there is a slight decrease of OOD robustness as the models are fine-tuned. Results of the converged OOD robustness can be found in [Section 8.2](#). We observe distinct patterns in robustness across both fine-tuning methods (i.e., mainly between PEFTs and traditional fine-tuning methods) and training domains. Linear probing consistently yields the lowest OOD robustness scores ( $61\% \pm 5\%$ ) across all domains, while full fine-tuning demonstrates superior robustness ( $73\% \pm 2\%$ ). Notably, the "real" domain exhibits substantially lower robustness scores ( $64\% \pm 5\%$ ) compared to other domains such as "infograph" ( $0.73\% \pm 4\%$ ) and "quickdraw" ( $0.72\% \pm 3\%$ ). This phenomenon can be attributed to the relatively more significant distribution shift of domains other than "real" from the pre-training dataset. Additionally, OOD robustness scores for all training domains are consistent across the five different PEFTs with  $\sim 0.03$  standard deviation, suggesting the parameter-efficient strategies can support OOD generalization to a similar degree.

In summary, these empirical results suggest that while adversarial robustness and OOD robustness both measure capabilities of models fine-tuned by different PEFTs on images outside their standard test datasets, they are driven by different underlying logics. OOD robustness emphasizes generalization across distributions and has trends aligning well with those of standard accuracy, whereas adversarial robustness focuses on specific perturbations and is in conflict with standard accuracy after certain point of fine-tuning process. This finding has important implications for designing fine-tuning recipes (i.e., both mechanisms and training procedures) that are both safe and robust in real-world applications.

## 5 DISCUSSION & LIMITATIONS

---

The key focus of this thesis is to explore how the shift to using parameter-efficient fine-tuning strategies impact the robustness-accuracy trade-off space. This is motivated by that, based on the literature in the field for the recent three years, people have been designing new PEFTs for higher accuracy and performance (see [chapter 6](#)) but with little attention to integrating them with previously proposed approaches to improve robustness of the transfer learning pipeline, not to mention studying on how each different PEFT techniques differ in robustness, especially, the robustness-accuracy trade-off space, in security and safety settings. Furthermore, adversarial training has been proposed to be crucial for achieving higher robustness, while it is highly computational expensive and is not integrated to most off-the-shelf pre-trained models. We intend to bridge these gaps and come up with practical recipes/recommendations by evaluating the robustness-accuracy trade-off throughout the fine-tuning process with state-of-the-art PEFTs. We focus on the CV domain for this work, but we want to explore and extend the framework to other domains such as network intrusion detection and malware detection in the future. Some interesting directions and limitations are discussed below.

### 5.1 Defining the Space of PEFTs

We acknowledge the size of the space of parameter-efficient fine-tuning strategies. In order to probe the PEFT space and study how different components affect robustness-accuracy trade-offs, we have to take parameter-efficient fine-tuning strategies, training schemes, and their application domains into consideration.

**PEFT Strategies.** As mentioned before, there are two key dimensions of PEFTs: information extracted and underlying mechanisms. In our experi-

ments, we study and vary five representative state-of-the-art PEFTs from three major categories together with full fine-tuning and linear probing to probe the space. However, we are aware that there are new strategies being proposed and could be integrated into our experimental pipeline in the future. Furthermore, there are many other factors to be considered, such as the number of trainable parameters and combinations PEFT strategies. Compacter is the only hybrid strategy that we considered in the experiments, and we plan to integrate more in the future.

**Training schemes.** Two stages of training are relevant here—(1) training the pre-trained models and (2) adapting the pre-trained model with PEFTs during fine-tuning. We use the ViT-Base model from HuggingFace, which is trained using supervised learning on ImageNet21k, and we leave CLIP models (Contrastive Language-Image Pretraining) which uses self-supervised learning to future work in order to see if the same trend applies to different pre-training schemes. It is worth noting that CLIP models show similar results as those of supervised-trained models based on other literature [31, 56]. In terms of fine-tuning schemes, we would like to extend this work with customized adversarial training integrated to study its impact on the trade-off space.

**Domains.** PEFT strategies are typically applied to models with common architectures, such as transformer-based models, whose key attention components are consistent across domains like CV, NLP, and speech recognition. This makes our experimental framework more adaptable to different domains. However, the tasks across these domains (i.e., variations in data formats, sizes, and objectives) differ significantly. Given that model robustness is closely related to task complexity, as our findings suggest, it is crucial to consider these domain-specific differences while studying the balance between model accuracy and robustness. We leave adapting our framework for applications across various domains to future work.

## 5.2 Adversarial Training

Adversarial training has been extensively studied [28, 38] and is recognized as a crucial component for pre-training models intended for achieving high robustness on downstream tasks [21, 22, 23, 24]. However, as discussed in the thesis, the high computational cost associated with adversarial training often makes it impractical for most off-the-shelf pre-trained models, which are typically trained without adversarial methods. This limitation motivates our study on robustness-accuracy trade-offs using standard-trained pre-trained models. In comparison, incorporating adversarial training at the fine-tuning stage offers a promising alternative. During fine-tuning, only a small percentage of a model’s parameters are updated using a relatively small dataset, which tremendously reduces the computational burden of adversarial training. Our future work aims to explore more efficient methods for integrating adversarial training with PEFTs.

## 5.3 Security & Safety Measures

Moreover, we consider robustness in both security and safety settings for PEFTs. There is a diverse range of measures in terms of adversarial attacks and domain shifts. For adversarial robustness, we used one of the representative white-box evasion attack algorithms, while other ones can be easily integrated into our pipeline to assess how attack-dependent the model robustness is. Similarly, corrupted images can be used to extend the study on model safety robustness.



## 6 RELATED WORK

---

We discuss two main lines of related work here: (1) robustness of traditional transfer learning pipelines before the emergence of PEFTs, which evaluates and introduces new training schemes or architectural designs to improve the robustness of downstream models, and (2) advancement of hybrid PEFT methods, which explores the PEFT space by studying core components of the strategies for higher accuracy and less memory and computational requirements. However, as discussed, there is an under-explored gap between these two parallel lines of work—how does the advancement and limitations of the new PEFTs with various core mechanisms and transfer learning recipes impact robustness in practice.

### 6.1 Robustness of Traditional Transfer Learning

There are two crucial directions on the robustness of the traditional transfer learning pipeline—the pre-training phase and the fine-tuning phase. We focus on the latter here. In order to preserve the gained robustness of the pre-trained models during fine-tuning, researchers have introduced multiple branches of the model (several neural networks side-by-side) which are trained jointly with/without the same objective functions with pre-defined influence on each other. For example, TWINS [24] uses an adaptive net which is interconnected with the frozen net via batch normalization layers. Using low-rank adaptation, AutoLoRa [23] optimizes natural objective with a low-rank branch while optimizing adversarial objective with a standard feature extractor. In addition, [68] improves the learning rate scheduler to reduce overfitting induced by adversarial training during fine-tuning. However, to the best of our knowledge, there is no study focusing on evaluating, analyzing, and improving adversarial robustness of various PEFTs, not to mention the safety perspective of the

strategies. In this thesis, we extend the robustness metrics used for the traditional transfer learning pipeline.

We consider [56] as a closely related work, which starts to study the adversarial robustness of some PEFT methods and proposes a robustness-centered initialization technique for adversarial fine-tuning. However, they do not study how various PEFTs impacts the robustness-accuracy trade-off space, nor do they consider the safety perspective of the strategies.

## 6.2 Hybrid PEFT Methods

After the wide adoption of the PEFT strategies discussed previously, there are hybrid methods that study the PEFT space and employ a variety of approaches to combine strategies to further enhance model accuracy with lower computational costs. For example, UniPELT [69] decomposes LoRA, Prefix-tuning, and Adapters approaches into submodules and employs a *gating mechanism* to activate those PEFT submodules depending on input data and given tasks. Improving on the basic structure of the Adapter modules, Compacter [16] uses Kronecker products (a reparameterization technique) and shared weights to further reduce the size of the modules, which are then inserted into the original network. Another recent work [9] proposes the  $S_4$  design space, with which they group pre-trained model layers and assign different PEFT techniques to them for combination.

While these hybrid approaches explore the space of PEFT methods and their application schemes from different directions, they only use accuracy and computing resources as their evaluation metrics. No robustness impact is studied for the space, especially in the computer vision domain (after [31] adapt the standard PEFT methods from NLP to CV). We plan to extend our framework to include these hybrid methods in the future.

## 7 CONCLUSION

---

In this thesis, we explored the robustness-accuracy trade-off space of parameter-efficient fine-tuning strategies, addressing the misalignment between traditional transfer learning robustness studies and the practical robustness challenges of emerging PEFTs in security and safety contexts. We developed a systematic framework to decompose the training pipeline into pre-training, fine-tuning, and robustness evaluation phases, using adversarial examples and out-of-distribution data for dynamic assessment. Through extensive experiments, we fine-tuned and evaluated 231 models with 7 SOTA fine-tuning strategies across six datasets, performing over 4k robustness tests and analyzing their Pareto frontier curves. Our study first examined whether the robustness-accuracy trade-off persists in the pre-training and PEFT paradigm, and then analyzed the trade-off from two perspectives: (1) its sensitivity to PEFT strategies, backpropagation steps, and downstream tasks in adversarial settings, and (2) whether the findings extend to generalization robustness with OOD data.

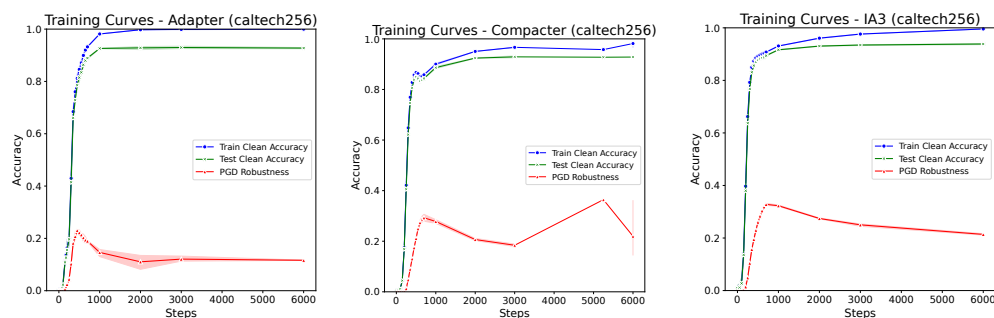
Our results show a consistent trade-off between adversarial robustness and standard accuracy at an early stage of fine-tuning across PEFT strategies and downstream tasks, but no significant trade-off in safety contexts. Moreover, the complexity and characteristics of downstream tasks emerge as critical factors in designing effective PEFT techniques and training recipes to achieve an optimal balance between robustness and accuracy. Importantly, security and safety robustness rely on distinct features of models, underscoring the need for independent evaluation to ensure comprehensive robustness assessments. This study provides actionable insights into designing and applying parameter-efficient fine-tuning strategies and their training recipes, offering guidelines for practitioners to enhance model robustness in both security and safety contexts while minimizing the compromise of standard accuracy.

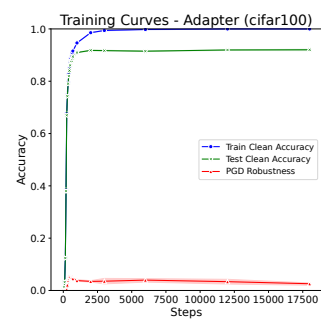
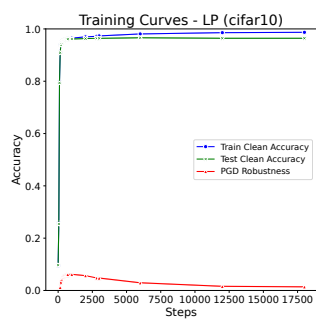
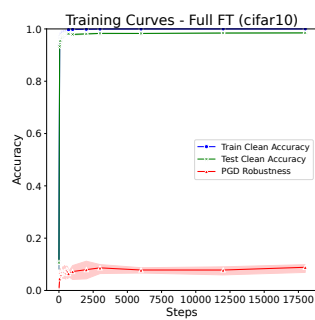
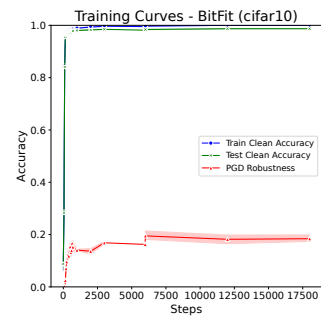
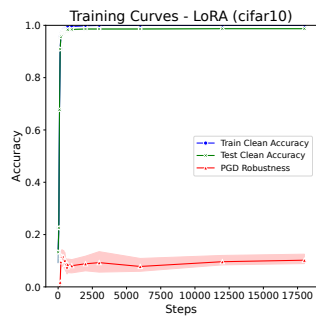
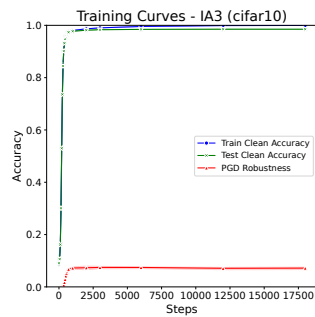
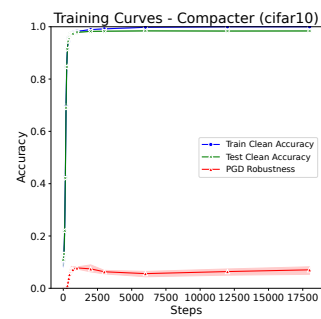
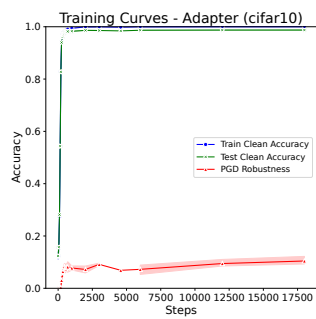
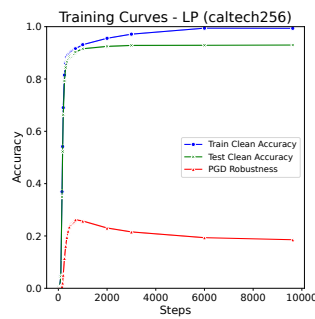
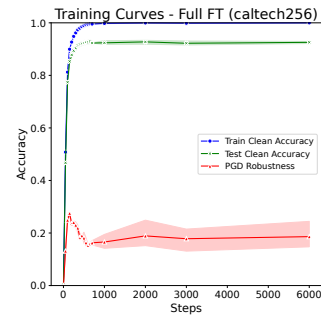
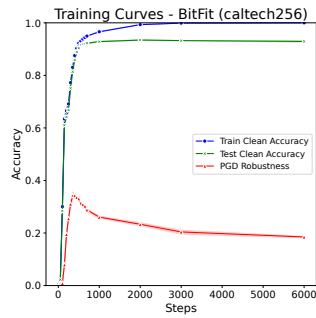
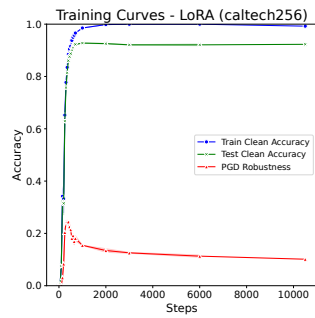
## 8 APPENDIX

Here, we present complete empirical results on our three research questions across seven fine-tuning strategies and six benchmark datasets.

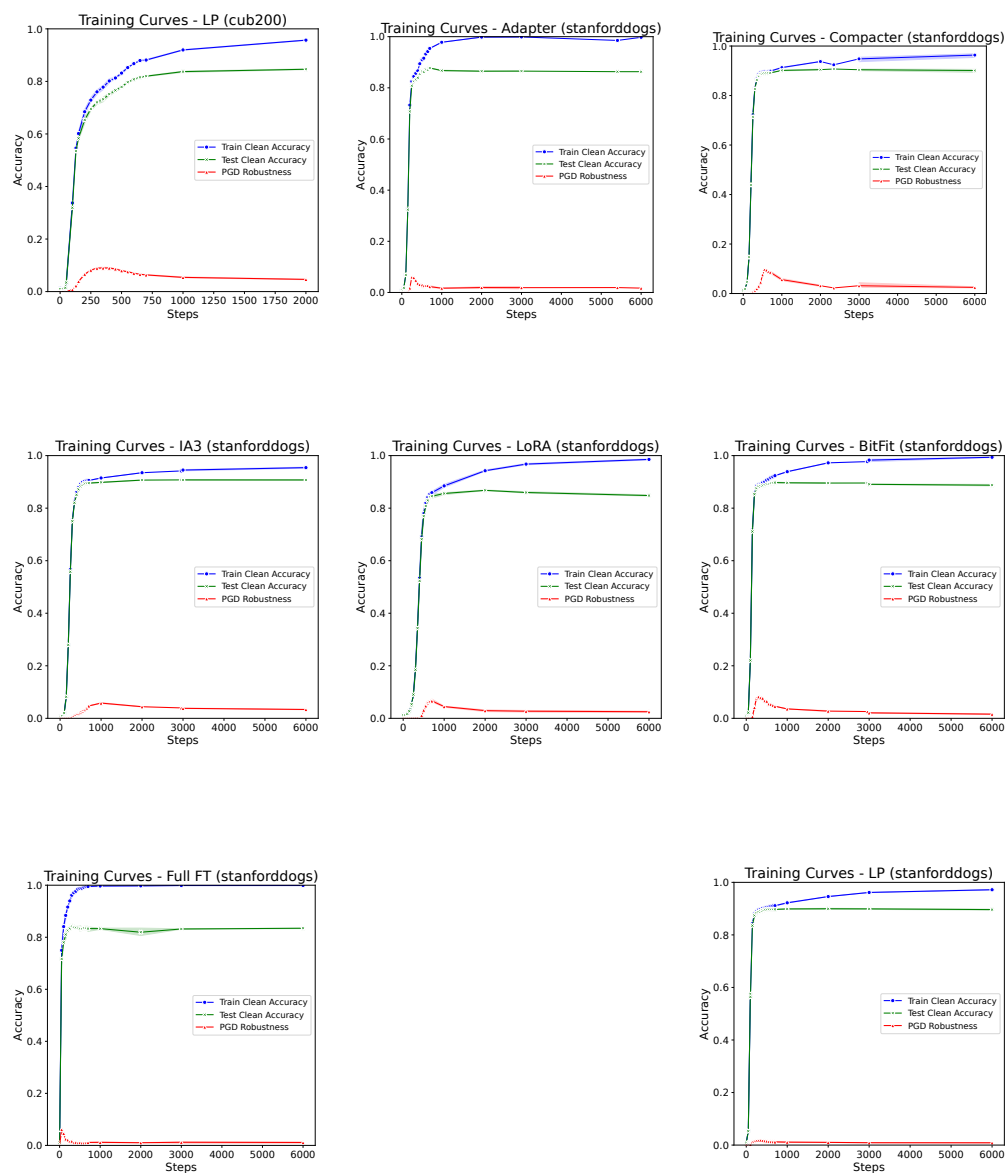
### 8.1 Training Curves with Adversarial Robustness

Full results of the trends in standard accuracy and adversarial robustness during training and testing (across the number of backpropagations) are shown in [Figure 8.1](#). Models are adapted by seven fine-tuning strategies on five datasets—Caltech256, CIFAR10, CIFAR100, CUB-200-2011, and Stanford Dogs. The trends are mostly consistent across all datasets and fine-tuning strategies as discussed in [Section 4.2](#). Note there is an anomaly increase for Compacter on Caltech256 at step 3000, we attribute this to an artifact issue and will update the results.









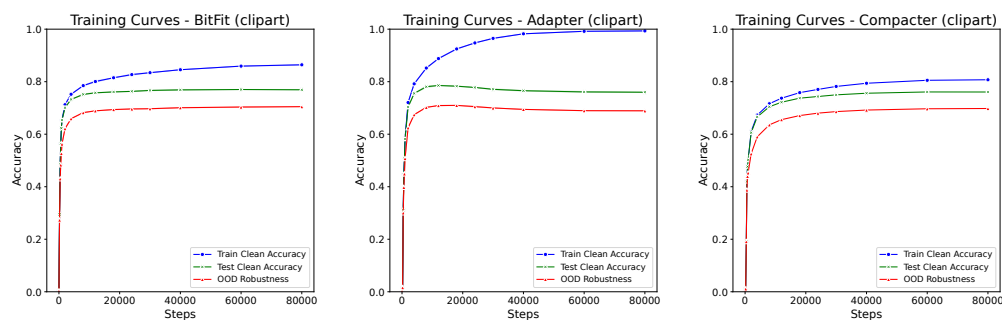
**Figure 8.1: The trend of training standard accuracy (blue), test standard accuracy (green), and adversarial robustness on test dataset (red) across the number of backpropagation steps on five datasets. PGD robustness reaches its peak and drops at an early stage of training, while both training and test clean accuracy keep increasing and plateau in the end.**

## 8.2 More Results on OOD Robustness

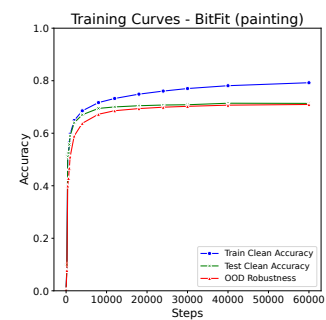
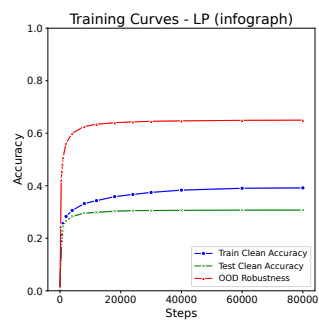
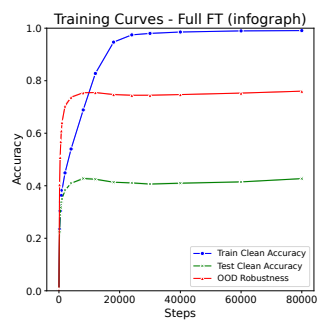
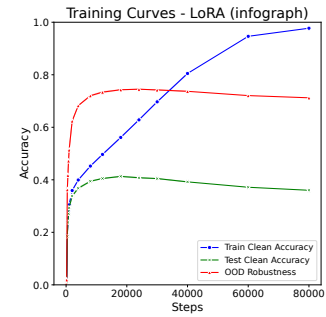
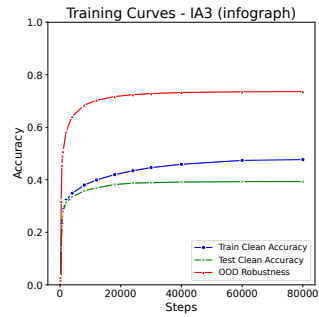
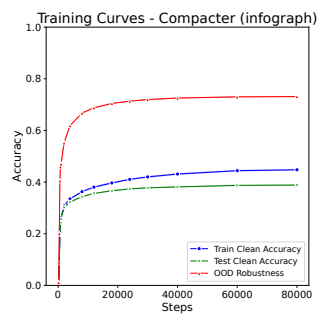
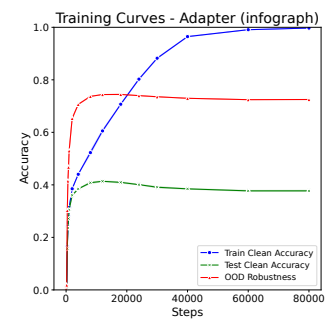
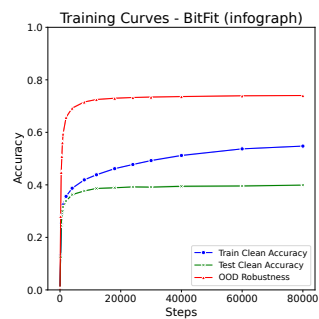
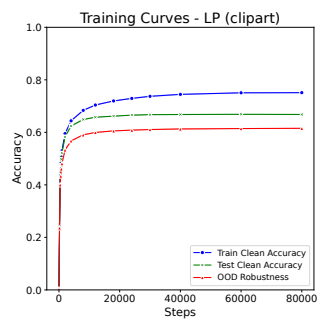
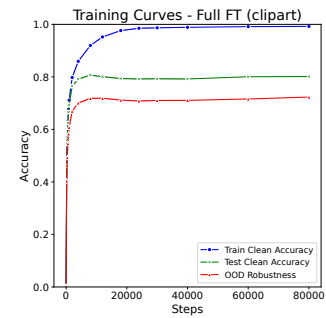
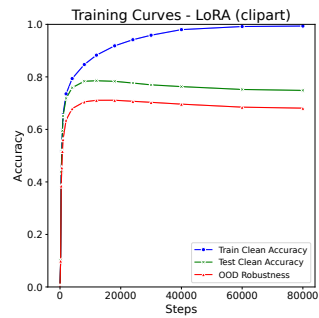
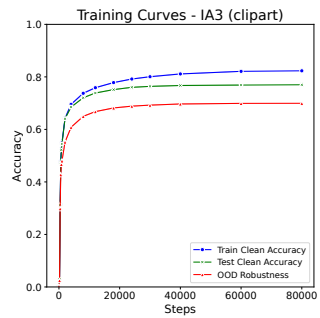
### Training Curves

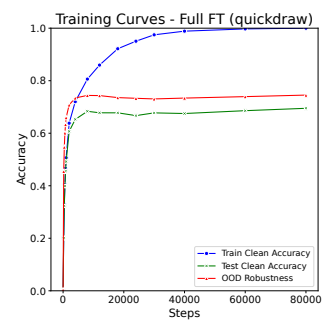
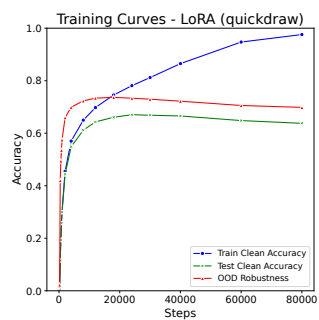
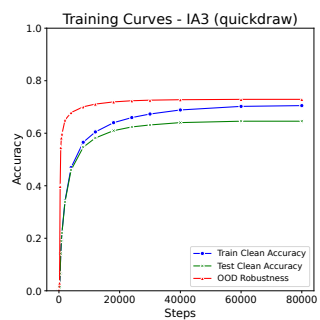
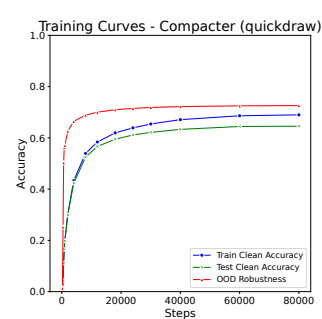
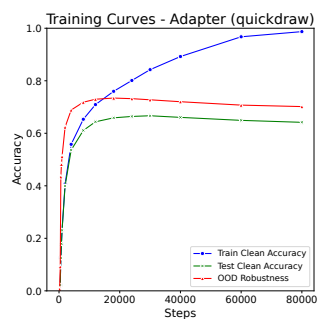
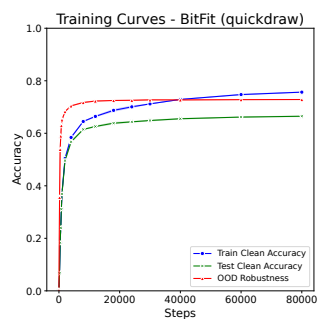
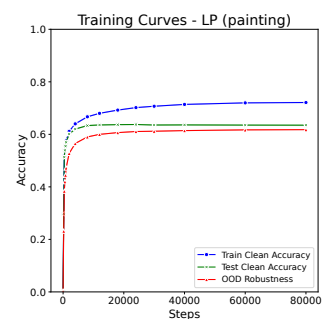
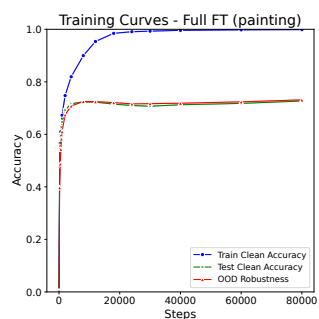
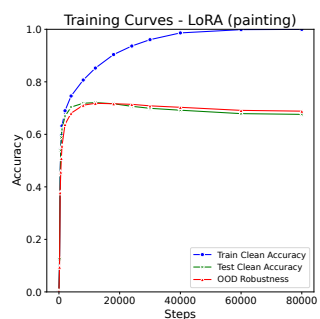
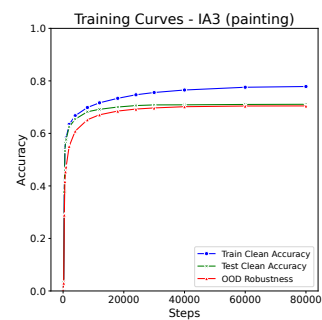
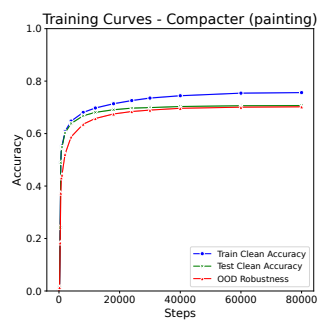
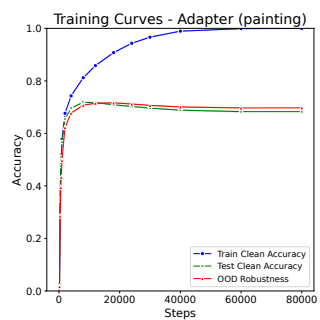
Full results of the trends in standard accuracy and OOD robustness during training and testing (across the number of backpropagations) are shown in [Figure 8.2](#). These results span six distinct domains—clip art, infograph, painting, quick draw, real, and sketch—as described in DomainNet [1]. The observed patterns are consistent with the discussion in [Section 4.4](#).

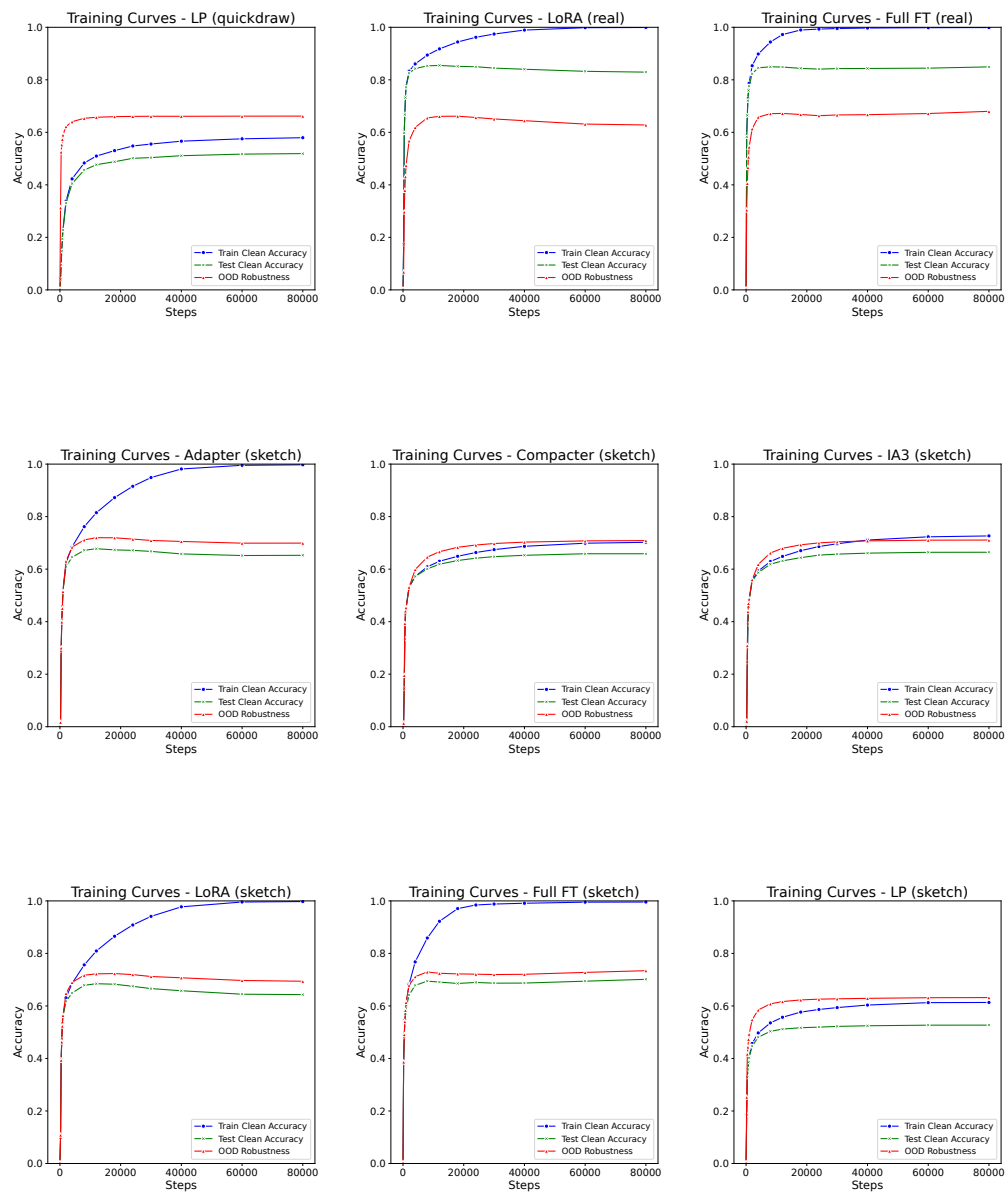
Interestingly, there are instances where robustness in OOD test data surpasses training or test accuracy. This phenomenon can be attributed to the domain of the pre-training data. The model is pre-trained on ImageNet-21k, which predominantly contains real images, where domains like infograph differ significantly. Even when the model is fine-tuned on infograph images, this fine-tuning may not provide sufficient detail for complete adaptation. As a result, the model may still regard real images as "in-domain". Thus, in the pre-training and fine-tuning paradigm, true OOD robustness should be evaluated on domains absent from both the pre-training and fine-tuning datasets.



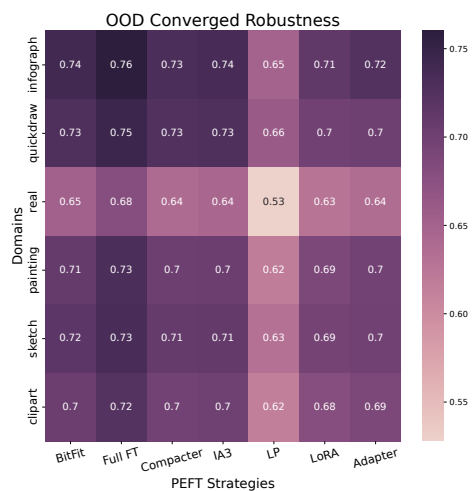








**Figure 8.2: The trend of training standard accuracy (blue), test standard accuracy in the training domain (green), and OOD robustness in other domains (red) across the number of backpropagation steps. The results of six training domains are shown here.**



**Figure 8.3: A heatmap of the converged OOD robustness towards the end of the fine-tuning phase across 6 domains.**

## Converged OOD Robustness

The heatmap ([Figure 8.3](#)) shows results of OOD robustness after it converges at the end of fine-tuning. The pattern observed is consistent as it is discussed in [Section 4.4](#).

BIBLIOGRAPHY

---

- [1] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, "Moment Matching for Multi-Source Domain Adaptation," Aug. 2019, arXiv:1812.01754 [cs]. [Online]. Available: <http://arxiv.org/abs/1812.01754>
- [2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," Aug. 2023, arXiv:1706.03762 [cs]. [Online]. Available: <http://arxiv.org/abs/1706.03762>
- [3] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," Jun. 2021, arXiv:2010.11929 [cs]. [Online]. Available: <http://arxiv.org/abs/2010.11929>
- [4] L. Dong, S. Xu, and B. Xu, "Speech-Transformer: A No-Recurrence Sequence-to-Sequence Model for Speech Recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2018, pp. 5884–5888, iSSN: 2379-190X. [Online]. Available: <https://ieeexplore.ieee.org/document/8462506>
- [5] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language Models are Few-Shot Learners," Jul. 2020, arXiv:2005.14165 [cs]. [Online]. Available: <http://arxiv.org/abs/2005.14165>

- [6] Y. Tay, M. Dehghani, J. Rao, W. Fedus, S. Abnar, H. W. Chung, S. Narang, D. Yogatama, A. Vaswani, and D. Metzler, "Scale Efficiently: Insights from Pre-training and Fine-tuning Transformers," Jan. 2022, arXiv:2109.10686 [cs]. [Online]. Available: <http://arxiv.org/abs/2109.10686>
- [7] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-Rank Adaptation of Large Language Models," Oct. 2021, arXiv:2106.09685 [cs]. [Online]. Available: <http://arxiv.org/abs/2106.09685>
- [8] X. L. Li and P. 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)*. Online: Association for Computational Linguistics, 2021, pp. 4582–4597. [Online]. Available: <https://aclanthology.org/2021.acl-long.353>
- [9] J. Chen, A. Zhang, X. Shi, M. Li, A. Smola, and D. Yang, "Parameter-Efficient Fine-Tuning Design Spaces," Jan. 2023, arXiv:2301.01821 [cs]. [Online]. Available: <http://arxiv.org/abs/2301.01821>
- [10] A. Edalati, M. Tahaei, I. Kobyzev, V. P. Nia, J. J. Clark, and M. Rezagholizadeh, "KronA: Parameter Efficient Tuning with Kronecker Adapter," Dec. 2022, arXiv:2212.10650 [cs]. [Online]. Available: <http://arxiv.org/abs/2212.10650>
- [11] D. Guo, A. M. Rush, and Y. Kim, "Parameter-Efficient Transfer Learning with Diff Pruning," Jun. 2021, arXiv:2012.07463 [cs]. [Online]. Available: <http://arxiv.org/abs/2012.07463>
- [12] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. de Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, "Parameter-Efficient Transfer Learning for NLP," Jun. 2019, arXiv:1902.00751 [cs, stat]. [Online]. Available: <http://arxiv.org/abs/1902.00751>
- [13] Q. V. Le, T. Sarlos, and A. J. Smola, "Fastfood: Approximate Kernel Expansions in Loglinear Time," Aug. 2014, arXiv:1408.3060 [cs, stat]. [Online]. Available: <http://arxiv.org/abs/1408.3060>

- [14] B. Lester, R. Al-Rfou, and N. Constant, "The Power of Scale for Parameter-Efficient Prompt Tuning," Sep. 2021, arXiv:2104.08691 [cs]. [Online]. Available: <http://arxiv.org/abs/2104.08691>
- [15] H. Liu, D. Tam, M. Muqeeth, J. Mohta, T. Huang, M. Bansal, and C. Raffel, "Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning," Aug. 2022, arXiv:2205.05638 [cs]. [Online]. Available: <http://arxiv.org/abs/2205.05638>
- [16] R. K. Mahabadi, J. Henderson, and S. Ruder, "Efficient Low-Rank Hypercomplex Adapter Layers."
- [17] E. B. Zaken, S. Ravfogel, and Y. Goldberg, "BitFit: Simple Parameter-efficient Fine-tuning for Transformer-based Masked Language-models," Sep. 2022, arXiv:2106.10199 [cs]. [Online]. Available: <http://arxiv.org/abs/2106.10199>
- [18] L. Rasmy, Y. Xiang, Z. Xie, C. Tao, and D. Zhi, "Med-BERT: pre-trained contextualized embeddings on large-scale structured electronic health records for disease prediction," May 2020, arXiv:2005.12833. [Online]. Available: <http://arxiv.org/abs/2005.12833>
- [19] K. Chitta, A. Prakash, B. Jaeger, Z. Yu, K. Renz, and A. Geiger, "TransFuser: Imitation with Transformer-Based Sensor Fusion for Autonomous Driving," May 2022, arXiv:2205.15997. [Online]. Available: <http://arxiv.org/abs/2205.15997>
- [20] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, "Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study," *Journal of Information Security and Applications*, vol. 50, p. 102419, Feb. 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2214212619305046>
- [21] L. Rice, E. Wong, and J. Z. Kolter, "Overfitting in adversarially robust deep learning," Mar. 2020, arXiv:2002.11569 [cs]. [Online]. Available: <http://arxiv.org/abs/2002.11569>

- [22] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry, "Adversarial Examples Are Not Bugs, They Are Features," Aug. 2019, arXiv:1905.02175. [Online]. Available: <http://arxiv.org/abs/1905.02175>
- [23] X. Xu, J. Zhang, and M. Kankanhalli, "AutoLoRa: A Parameter-Free Automated Robust Fine-Tuning Framework," Oct. 2023, arXiv:2310.01818 [cs]. [Online]. Available: <http://arxiv.org/abs/2310.01818>
- [24] Z. Liu, Y. Xu, X. Ji, and A. B. Chan, "TWINS: A Fine-Tuning Framework for Improved Transferability of Adversarial Robustness and Generalization," Mar. 2023, arXiv:2303.11135 [cs]. [Online]. Available: <http://arxiv.org/abs/2303.11135>
- [25] "google/vit-base-patch16-224-in21k · Hugging Face." [Online]. Available: <https://huggingface.co/google/vit-base-patch16-224-in21k>
- [26] R. Wightman, "Pytorch image models," <https://github.com/rwightman/pytorch-image-models>, 2019.
- [27] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning Transferable Visual Models From Natural Language Supervision," Feb. 2021, arXiv:2103.00020 [cs]. [Online]. Available: <http://arxiv.org/abs/2103.00020>
- [28] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards Deep Learning Models Resistant to Adversarial Attacks," Sep. 2019, arXiv:1706.06083 [cs, stat]. [Online]. Available: <http://arxiv.org/abs/1706.06083>
- [29] X. Zhai, J. Puigcerver, A. Kolesnikov, P. Ruysen, C. Riquelme, M. Lucic, J. Djolonga, A. S. Pinto, M. Neumann, A. Dosovitskiy, L. Beyer, O. Bachem, M. Tschannen, M. Michalski, O. Bousquet, S. Gelly, and N. Houlsby, "A Large-scale Study of Representation Learning with the Visual Task Adaptation Benchmark," Feb. 2020, arXiv:1910.04867 [cs, stat]. [Online]. Available: <http://arxiv.org/abs/1910.04867>



- [30] A. Kumar, A. Raghunathan, R. Jones, T. Ma, and P. Liang, "Fine-Tuning can Distort Pretrained Features and Underperform Out-of-Distribution," Feb. 2022, arXiv:2202.10054 [cs]. [Online]. Available: <http://arxiv.org/abs/2202.10054>
- [31] X. He, C. Li, P. Zhang, J. Yang, and X. E. Wang, "Parameter-efficient Model Adaptation for Vision Transformers," Jul. 2023, arXiv:2203.16329 [cs]. [Online]. Available: <http://arxiv.org/abs/2203.16329>
- [32] V. Lialin, V. Deshpande, and A. Rumshisky, "Scaling Down to Scale Up: A Guide to Parameter-Efficient Fine-Tuning," Mar. 2023, arXiv:2303.15647 [cs]. [Online]. Available: <http://arxiv.org/abs/2303.15647>
- [33] Y.-L. Sung, J. Cho, and M. Bansal, "LST: Ladder Side-Tuning for Parameter and Memory Efficient Transfer Learning," Oct. 2022, arXiv:2206.06522 [cs]. [Online]. Available: <http://arxiv.org/abs/2206.06522>
- [34] J. Pfeiffer, A. Kamath, A. Rücklé, K. Cho, and I. Gurevych, "AdapterFusion: Non-Destructive Task Composition for Transfer Learning," Jan. 2021, arXiv:2005.00247 [cs]. [Online]. Available: <http://arxiv.org/abs/2005.00247>
- [35] Y.-L. Sung, V. Nair, and C. Raffel, "Training Neural Networks with Fixed Sparse Masks."
- [36] A. Aghajanyan, L. Zettlemoyer, and S. Gupta, "Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning," Dec. 2020, arXiv:2012.13255 [cs]. [Online]. Available: <http://arxiv.org/abs/2012.13255>
- [37] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and Harnessing Adversarial Examples," Mar. 2015, arXiv:1412.6572 [cs, stat]. [Online]. Available: <http://arxiv.org/abs/1412.6572>
- [38] N. Carlini and D. Wagner, "Towards Evaluating the Robustness of Neural Networks," Mar. 2017, arXiv:1608.04644 [cs]. [Online]. Available: <http://arxiv.org/abs/1608.04644>

- [39] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, May 2017. [Online]. Available: <https://dl.acm.org/doi/10.1145/3065386>
- [40] “STL-10 dataset.” [Online]. Available: <https://ai.stanford.edu/~acoates/stl10/>
- [41] A. Kurakin, I. Goodfellow, and S. Bengio, “Adversarial examples in the physical world,” Feb. 2017, arXiv:1607.02533 [cs, stat]. [Online]. Available: <http://arxiv.org/abs/1607.02533>
- [42] F. Croce and M. Hein, “Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks,” Aug. 2020, arXiv:2003.01690 [cs, stat]. [Online]. Available: <http://arxiv.org/abs/2003.01690>
- [43] A. Krizhevsky, “Learning Multiple Layers of Features from Tiny Images.”
- [44] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar, “Do CIFAR-10 Classifiers Generalize to CIFAR-10?” Jun. 2018, arXiv:1806.00451 [cs, stat]. [Online]. Available: <http://arxiv.org/abs/1806.00451>
- [45] R. K. Mahabadi, J. Henderson, and S. Ruder, “Compacter: Efficient Low-Rank Hypercomplex Adapter Layers,” Nov. 2021, arXiv:2106.04647 [cs]. [Online]. Available: <http://arxiv.org/abs/2106.04647>
- [46] A. Y. Ng, “Feature selection,  $L_1$  vs.  $L_2$  regularization, and rotational invariance,” in *Twenty-first international conference on Machine learning - ICML '04*. Banff, Alberta, Canada: ACM Press, 2004, p. 78. [Online]. Available: <http://portal.acm.org/citation.cfm?doid=1015330.1015435>
- [47] Y. Yao, L. Rosasco, and A. Caponnetto, “On Early Stopping in Gradient Descent Learning,” *Constructive Approximation*, vol. 26, no. 2, pp. 289–315, Aug. 2007. [Online]. Available: <http://link.springer.com/10.1007/s00365-006-0663-2>

- [48] P. Nakkiran, G. Kaplun, Y. Bansal, T. Yang, B. Barak, and I. Sutskever, “Deep Double Descent: Where Bigger Models and More Data Hurt,” Dec. 2019, arXiv:1912.02292. [Online]. Available: <http://arxiv.org/abs/1912.02292>
- [49] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: Nsga-ii,” *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [50] Y. Chen, K. Zhou, Y. Bian, B. Xie, B. Wu, Y. Zhang, K. Ma, H. Yang, P. Zhao, B. Han, and J. Cheng, “Pareto Invariant Risk Minimization: Towards Mitigating the Optimization Dilemma in Out-of-Distribution Generalization,” Mar. 2023, arXiv:2206.07766 [cs]. [Online]. Available: <http://arxiv.org/abs/2206.07766>
- [51] R. Sheatsley, E. Pauley, B. Hoak, and P. McDaniel, “The Space of Adversarial Strategies,” 2023.
- [52] D. Tsipras, S. Santurkar, L. Engstrom, A. Turner, and A. Madry, “Robustness May Be at Odds with Accuracy,” Sep. 2019, arXiv:1805.12152 [cs, stat]. [Online]. Available: <http://arxiv.org/abs/1805.12152>
- [53] C. of High Throughput Computing, “Chtc home.” [Online]. Available: <https://chtc.cs.wisc.edu/>
- [54] T. Ridnik, E. Ben-Baruch, A. Noy, and L. Zelnik-Manor, “ImageNet-21K Pretraining for the Masses,” Aug. 2021, arXiv:2104.10972 [cs]. [Online]. Available: <http://arxiv.org/abs/2104.10972>
- [55] “AdapterHub Documentation — AdapterHub documentation.” [Online]. Available: <https://docs.adapterhub.ml/index.html>
- [56] A. Hua, J. Gu, Z. Xue, N. Carlini, E. Wong, and Y. Qin, “Initialization Matters for Adversarial Transfer Learning,” Mar. 2024, arXiv:2312.05716 [cs]. [Online]. Available: <http://arxiv.org/abs/2312.05716>
- [57] “CIFAR-10 and CIFAR-100 datasets.” [Online]. Available: <https://www.cs.toronto.edu/~kriz/cifar.html>
- [58] G. Griffin, A. Holub, and P. Perona, “Caltech 256,” Apr 2022.

- [59] F.-F. Li, M. Andreeto, M. Ranzato, and P. Perona, "Caltech 101," Apr 2022.
- [60] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona, "Caltech-UCSD Birds 200," California Institute of Technology, Tech. Rep. CNS-TR-2010-001, 2010.
- [61] A. Khosla, N. Jayadevaprakash, B. Yao, and F.-F. Li, "Novel Dataset for Fine-Grained Image Categorization: Stanford Dogs."
- [62] Y. Li and C. Xu, "Trade-off between Robustness and Accuracy of Vision Transformers," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Vancouver, BC, Canada: IEEE, Jun. 2023, pp. 7558–7568. [Online]. Available: <https://ieeexplore.ieee.org/document/10205096/>
- [63] D. Su, H. Zhang, H. Chen, J. Yi, P.-Y. Chen, and Y. Gao, "Is Robustness the Cost of Accuracy? – A Comprehensive Study on the Robustness of 18 Deep Image Classification Models," Mar. 2019, arXiv:1808.01688 [cs]. [Online]. Available: <http://arxiv.org/abs/1808.01688>
- [64] A. Raghunathan, J. Steinhardt, and P. Liang, "Certified Defenses against Adversarial Examples," Oct. 2020, arXiv:1801.09344. [Online]. Available: <http://arxiv.org/abs/1801.09344>
- [65] H. Zhang, Y. Yu, J. Jiao, E. P. Xing, L. E. Ghaoui, and M. I. Jordan, "Theoretically Principled Trade-off between Robustness and Accuracy," Jun. 2019, arXiv:1901.08573 [cs]. [Online]. Available: <http://arxiv.org/abs/1901.08573>
- [66] A. Torralba and A. A. Efros, "Unbiased look at dataset bias," in *CVPR 2011*, Jun. 2011, pp. 1521–1528, iSSN: 1063-6919. [Online]. Available: <https://ieeexplore.ieee.org/document/5995347>
- [67] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.
- [68] A. Jeddi, M. J. Shafiee, and A. Wong, "A Simple Fine-tuning Is All You Need: Towards Robust Deep Learning Via Adversarial Fine-tuning," Dec. 2020, arXiv:2012.13628 [cs]. [Online]. Available: <http://arxiv.org/abs/2012.13628>

- [69] Y. Mao, L. Mathias, R. Hou, A. Almahairi, H. Ma, J. Han, W.-t. Yih, and M. Khabsa, "UniPELT: A Unified Framework for Parameter-Efficient Language Model Tuning," Sep. 2022, arXiv:2110.07577 [cs]. [Online]. Available: <http://arxiv.org/abs/2110.07577>