ADAPT to Robustify Prompt Tuning Vision Transformers

Anonymous authors
Paper under double-blind review

Abstract

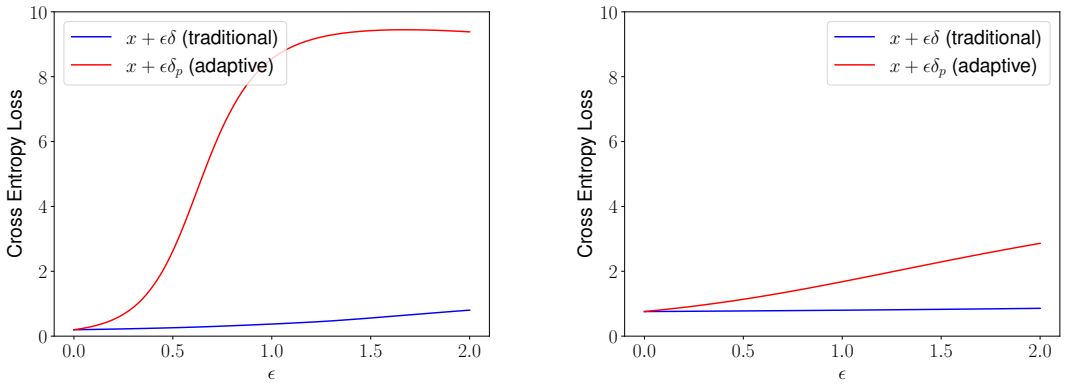
The performance of deep models, including Vision Transformers, is known to be vulnerable to adversarial attacks. Many existing defenses against these attacks, such as adversarial training, rely on full-model fine-tuning to induce robustness in the models. These defenses require storing a copy of the entire model, that can have billions of parameters, for each task. At the same time, parameter-efficient prompt tuning is used to adapt large transformer-based models to downstream tasks without the need to save large copies. In this paper, we examine parameter-efficient prompt tuning of Vision Transformers for downstream tasks under the lens of robustness. We show that previous adversarial defense methods, when applied to the prompt tuning paradigm, suffer from *gradient obfuscation* and are vulnerable to adaptive attacks. We introduce ADAPT, a novel framework for performing adaptive adversarial training in the prompt tuning paradigm. Our method achieves competitive robust accuracy of $\sim 40\%$ w.r.t. SOTA robustness methods using full-model fine-tuning, by tuning only $\sim 1\%$ of the number of parameters.

1 Introduction

Despite their success, deep neural networks, including the popular ViT architecture, have been shown to be vulnerable to adversarial attacks (Szegedy et al., 2013; Mahmood et al., 2021). These attacks are modifications to input images, which are generally imperceptible to humans but can drastically change the prediction of a neural network. Many countermeasures have been introduced to induce robustness against such attacks, one common approach being adversarial training (Madry et al., 2017), where the input is augmented with said perturbations during training. While originally introduced for CNNs, adversarial training has been extended to ViTs in the full-model fine-tuning paradigm (Mo et al., 2022). However, adversarial training and similar methods are computationally expensive and difficult to perform for large datasets as it requires performing multi-step evaluation of adversarial examples. Hence ViTs are generally pre-trained without such considerations.

Transformer-based language models (Brown et al., 2020) which can consist of billions of parameters require pretraining on massive amounts of data. To fine-tune such models for different tasks, one would have to save a separate copy of the weights per task, making it highly inefficient in terms of parameters. However, it is shown that these models can be adapted to downstream tasks with fewer parameters, for example using prompt design (Brown et al., 2020) or prompt tuning (Lester et al., 2021). In prompt tuning, optimizable tokens are prepended to the model input in order to leverage the pretraining knowledge and adapt to the downstream task without changing the weights of the model. This allows for a lightweight adaptation of these large models. ViTs can benefit from the same technique (Jia et al., 2022), especially as their parameters continue to increase (Dehghani et al., 2023).

It is therefore important to explore the utility of prompt tuning for the adversarial robustness of ViTs. In this paper, we show that existing adversarial defense methods, when applied to the prompt tuning paradigm, suffer from *gradient obfuscation* (Athalye et al., 2018). This means the gradients of the model are not useful for obtaining adversarial perturbations, and training using existing methods will lead to a *false sense of security*. We demonstrate this phenomenon empirically using single-step attacks, and then design an adaptive attack to expose the vulnerability of existing methods. Finally, we propose a framework for **AD**aptive



(a) Adversarial Training + Prompt Tuning

(b) ADAPT$_{CE}$ + Prompt Tuning

Figure 1: **Existing methods do not exhibit robustness to adaptive attacks.** Comparison of the cross entropy loss of a random batch along two adversarial perturbation directions. Left (a) depicts the loss values of a prompt trained with traditional adversarial training and right (b) shows the loss values of a prompt trained with ADAPT$_{CE}$. Adversarial Training + Prompt Tuning does not exhibit a significant increase in the loss under the traditional PGD direction, but shows significant vulnerability to adaptive attacks. In contrast, the proposed method ADAPT$_{CE}$, exhibits robustness to both perturbation directions.

Adversarial **P**rompt **T**uning (ADAPT) to overcome this vulnerability. We empirically demonstrate the superior robustness of our method in multiple settings.

To the best of our knowledge, we are the first to study this issue extensively. A summary of our contributions is as follows:

- We investigate the adversarial robustness of the prompt tuning paradigm for ViTs. We show that the existing methods, when applied to the prompt tuning scenario, suffer from gradient obfuscation (Athalye et al., 2018).
- We design an adaptive attack accordingly and show that existing methods are significantly vulnerable to our attack, resulting in a drop in their robust accuracy to approximately 1% (See fig. 1).
- We propose a novel loss function that emphasizes conditioning on the prompts to craft adversarial examples during training. We quantitatively demonstrate the superiority of our method (ADAPT), in achieving robustness compared to existing methods.

2 Related Works

Vision Transformers. Inspired by the groundbreaking success of Transformers for Natural Language Processing (NLP) tasks (Vaswani et al., 2017; Kenton & Toutanova, 2019; OpenAI, 2023), Vision Transformers (ViTs) (Dosovitskiy et al., 2020) are an adaptation of the architecture for computer vision tasks. Similar to the 'tokens' fed to NLP transformers, here the input images are split into fixed-size patches, which are then embedded and fed to the transformer layers of the ViT. ViTs are powerful architectures and achieve state-of-the-art performance on several downstream vision tasks (Touvron et al., 2021; Caron et al., 2021; He et al., 2022).

Prompting. This technique was introduced in the NLP literature to eliminate the need to fine-tune large pre-trained language models for downstream tasks (Liu et al., 2023). Intuitively, prompting reformulates the downstream task so that it closely resembles the task that a frozen model has been pre-trained to solve. Prompt tuning (Lester et al., 2021), *i.e.*, training the prompts for downstream tasks, has been extended to

ViTs and is highly effective for domain generalization (Zheng et al., 2022), continual learning (Wang et al., 2022), and several other vision tasks.

Adversarial Robustness. Adversarial Training (AT) was first introduced for CNNs and was thoroughly investigated (Szegedy et al., 2013; Goodfellow et al., 2014; Madry et al., 2017) with several variations (Buckman et al., 2018; Zhang et al., 2019; 2020). ViTs have also been shown to be vulnerable to adversarial attacks (Mahmood et al., 2021). However, AT and its variants are computationally expensive due to requiring multiple forward and backward passes to obtain adversarial examples to train with. This makes them difficult to use for pre-training ViTs. Therefore, adversarial defense methods are typically only adopted during fine-tuning for downstream tasks (Mo et al., 2022).

Robust Prompts. Prompt tuning has been employed for enhancing the robustness of NLP transformers (Yang et al., 2022), which deal with discrete inputs. Extending the same robustness to the continuous input space for ViTs is non-trivial and has not been explored in the literature.

There are other works which explore robustness of visual prompt tuning (Chen et al., 2023a). However, they are not tailored to the transformer paradigm. Moreover, their number of tuned parameters grows with the number of classes in the dataset. Visual prompting or adversarial reprogramming was proposed before the advent of prompt tuning for adapting CNNs to downstream tasks (Elsayed et al., 2018). As such, we are the first to investigate robust prompting for vision transformers.

3 Notations and Preliminaries

3.1 Vision Transformer

The Vision Transformer (Dosovitskiy et al., 2020) has proven to be quite effective at performing various computer vision tasks. For our purposes, we consider the classification task. A transformer $f_{\Phi,\Psi}$ as a classifier consists of a feature extractor Φ and a linear classifier Ψ operating on the extracted features. That is,

$$\Phi : \mathcal{X} \to \mathbb{R}^d, \Psi : \mathbb{R}^d \to \mathbb{R}^{|\mathcal{Y}|} \tag{1}$$

$$f_{\Phi,\Psi}(x) = softmax(\Psi(\Phi(x))) \tag{2}$$

where $\mathcal{X}$ is the input space, $\mathcal{Y}$ the label space, $|\mathcal{Y}|$ is the number of classes, and d is the hidden dimension of the transformer (which stays the same throughout its layers). As the transformer architecture operates on inputs that are sequences of high-dimensional vectors, we consider the input space of the transformer to be the space of all d-dimensional (the hidden dimension of the transformer) sequences of arbitrary length k:

$$\mathcal{X} := \bigcup_{\forall k \in \mathbb{N}} \mathbb{R}^{d \times k} \tag{3}$$

As such, to transfer images from the data distribution support, i.e. $\mathbb{R}^{h \times w \times c}$, to the transformer input space $\mathcal{X}$, the image is divided into n non-overlapping patches and then transformed via a linear transformation or through a convolutional neural network (CNN). Formally, we have

$$T : \mathbb{R}^{h \times w \times c} \to \mathbb{R}^{d \times n} \tag{4}$$

as the embedding function. For the sake of brevity, except in sections where the embedding function is being discussed directly, we omit T from our notation and assume that inputs are being passed through it.

3.2 Prompt Tuning

With the advent of transformers and large language models with billions of parameters pre-trained on extremely large amounts of data, it is desirable to tune these models for downstream tasks at a minimal cost. While full-model fine-tuning is an effective option, it requires training and storing an entirely new set of

model weights for each downstream task, which becomes inefficient in terms of memory as both the number of tasks and the number of model parameters grow. A simple yet effective solution to this is prompt tuning (PT) (Lester et al., 2021), which proceeds by simply appending tunable prompt tokens to the transformer input space. Depending on the model size and number of tokens, it can yield comparable performance to full-model fine-tuning. This allows us to adapt to downstream tasks without having to store copies of extremely large models, and instead only storing parameter-efficient prompts.

Formally, a pre-trained Transformer f_{Φ^*,Ψ^*} is obtained by training on large amounts of data, *i.e.*,

$$\Phi^*, \Psi^* = \arg\min_{\Phi,\Psi} \mathbb{E}_{x,y\sim P}\mathcal{L}(f_{\Phi,\Psi}(x), y). \tag{5}$$

For supervised learning, this can be solved using Empirical Risk Minimisation (ERM), where P is a data distribution used for pre-training, e.g. ImageNet, and $\mathcal{L}$ is the loss function used for training. The goal is then to adapt to a downstream data distribution D given Φ^*.

Full-model fine-tuning (which we will refer to as Fine-Tuning (FT)) seeks to find some $\hat{\Phi}, \hat{\Psi}$ to minimize the expected loss $\mathbb{E}_{x,y\sim D}\ \mathcal{L}(f_{\hat{\Phi},\hat{\Psi}}(x), y)$ by using Φ^* as a starting point for $\hat{\Phi}$. Prompt tuning, instead, seeks to solve the following optimization problem

$$\hat{\theta}_p, \hat{\Psi} = \underset{\theta_p,\Psi}{argmin}\ \mathbb{E}_{x,y\sim D}\ \mathcal{L}(f(\theta_p, x), y) \tag{6}$$

Prompt tuning can be performed in multiple ways. In the case of simple Prompt Tuning (PT), a set of prompts is prepended to the input layer of the transformer. A prompted ViT $f(\theta_p, x)$ is defined as follows

$$f(\theta_p, x) := f_{\Phi^*,\Psi}([\theta_p, x]) \tag{7}$$

$$\theta_p \in \mathbb{R}^{d\times m} \tag{8}$$

where $[.,.]$ is the concatenation operator, d is the hidden dimension of the transformer, and m is the prompt length (recall that the input space of the transformer is arbitrary length sequences). Another way to perform prompt tuning is Prefix tuning (Li & Liang, 2021) or PT-v2 tuning (PT2) (Liu et al., 2021), where similar to PT, the backbone weights are kept unchanged. However, a set of prompt tokens are instead directly prepended to the input of *every* layer of the transformer, by only prepending to the key and value matrices of the self-attention mechanism (while keeping the same number of tokens per layer). That is, $f(\theta_p, x)$ in the case of PT2 is as follows

$$f(\theta_p, x) := f_{[\theta_p,\Phi^*],\Psi}(x) \tag{9}$$

$$\theta_p \in \mathbb{R}^{L\times d\times m} \tag{10}$$

where $[\theta_p; \Phi^*]$ is shorthand for prepending tokens to the input of every layer of the frozen feature extractor Φ^*, L is the number of transformer layers and m is the prompt length, *i.e.* the number of tokens tuned per layer.

3.3 Adversarial Robustness

Given an arbitrary classifier $f : \mathcal{X} \to \mathcal{Y}$, input $x \in \mathcal{X}$ with the true label $y \in \mathcal{Y}$, an adversary seeks to find some *imperceptible* perturbation δ with the same dimensionality as x, such that $x + \delta \in \mathcal{X}$ and $f(x + \delta) \neq y$. To achieve such imperceptibility, δ is typically constrained w.r.t. an ℓ_q norm, the most popular choices being $q = 2$ and $q = \infty$.

An untargetted adversarial attack for a classifier f bounded in ℓ_q norm by ϵ can be defined in the following way

$$x' = x + \underset{\|\delta\|_q\leq\epsilon}{argmax}\ \mathcal{L}(f(x + \delta), y) \tag{11}$$

The adversarial example defined above is an optimization problem that is typically solved using methods such as Projected Gradient Descent (PGD) (Madry et al., 2017). As deep learning models are highly susceptible

to such norm-bounded perturbations (Szegedy et al., 2013), it is crucial to defend against these attacks. However, it is important to note that PGD assumes full knowledge of the model parameters and inference algorithm, i.e. a *white-box* setting, for the adversary. Adversarial robustness can also be investigated through the lens of adversaries with limited knowledge, namely the *black-box* (Ilyas et al., 2018), and *grey-box* (Xu et al., 2021) settings. In this work, we focus on the white-box scenario.

One widely used defense method against adversarial attacks is Adversarial Training (Madry et al., 2017). If f_θ is a classifier parameterized by θ, AT tries to solve the following minimax optimization problem:

$$\min_{\theta} \ \mathbb{E}_{x,y\sim D}\left[\max_{\|\delta\|_q\leq\epsilon} \mathcal{L}(f_\theta(x+\delta),y)\right] \tag{12}$$

Practically, this objective trains the model on adversarially perturbed inputs with untargeted attacks.

3.3.1 Obfuscated Gradients.

An adversarial defense method suffers from *gradient masking* or *obfuscated gradients* (Athalye et al., 2018) when it does not have *"useful gradients"* for generating adversarial examples. Typically, there are three main causes for gradient obfuscation: exploding and vanishing gradients, stochastic gradients, and shattered gradients. We show that traditional adversarial defense methods in the prompt tuning paradigm suffer from shattered gradients. This means that the gradient of the model w.r.t. the input is incorrect and does not properly maximize the classification loss. To validate this claim, we use the observation of (Athalye et al., 2018) that in the presence of obfuscated gradients, single-step attacks are typically more effective than their multi-step counterparts.

4 Robust Prompt Tuning

In this section we introduce ADAPT, a novel framework for robustifying the parameter efficient prompt tuning of ViTs. We design an adaptive attack tailored to the prompt tuning paradigm by conditioning on the prompts during each gradient step. Using our adaptive attack formulation, we propose a novel loss function for adaptive robust training of prompts while fine tuning for downstream tasks.

4.1 Adaptive Adversarial Attack

In the prompt tuning scenario, $\nabla_x f(x)$ does not necessarily equal $\nabla_x f(\theta_p, x)$. Therefore, the existence of the prompt may lead to a form of gradient obfuscation (Athalye et al., 2018), *giving us a false sense of security* (See fig. 1a). In the experiment section , we demonstrate this effect by designing an adaptive attack and empirically show that while previous adversarial training methods may be robust against traditional attacks, they are vulnerable to our adaptive attack which reduces their robust accuracy to nearly zero.

In the prompt tuning paradigm, we reformulate the optimization objective for finding an adversarial example for classifier f as follows

$$x'_p = x + \underset{\|\delta_p\|_q\leq\epsilon}{argmax}\ \mathcal{L}_{CE}(f(\theta_p, x+\delta_p), y) \tag{13}$$

We can solve this optimization problem using Projected Gradient Descent (Madry et al., 2017) similar to eq. (11). Here, the key difference lies in conditioning on the prompt during each gradient step.

4.2 Prompt Adversarial Training

To perform adversarially robust prompt tuning, conditioning on the prompt is crucial for both generating the adversarial examples, as well as training the prompt itself.

As such, we propose a loss function for **AD**aptive **A**dversarial **P**rompt **T**uning (ADAPT)

$$\mathcal{L}_{ADAPT} = \mathcal{L}_p(f(\theta_p, x), y) + \lambda\mathcal{L}_{adv} \tag{14}$$

where $\mathcal{L}_p$ is the standard prompt tuning loss for any downstream task, and $\mathcal{L}_{adv}$ is the adversarial loss function which we will define below. Intuitively, the first term performs parameter-efficient fine-tuning by leveraging the information present in the frozen backbone, while the second term promotes robustness.

In this paper, we propose different choices for $\mathcal{L}_{adv}$:

Cross Entropy (CE). We can choose the cross-entropy loss for the prediction of the adaptive adversarial example:

$$\mathcal{L}_{adv} = \mathcal{L}_{CE}(f(\theta_p, x'_p), y) \tag{15}$$

which results in a loss function that promotes correct classification of samples perturbed with the adaptive attack.

KL Divergence. We can choose $\mathcal{L}_{adv}$ to be the KL-divergence between the prediction distributions of the unperturbed and perturbed samples:

$$\mathcal{L}_{adv} = \mathcal{L}_{KL}(f(\theta_p, x'_p), f(\theta_p, x)) \tag{16}$$

With this formulation, the first term in eq. (14) promotes the correct prediction of unperturbed samples, and the adversarial term eq. (16) matches the prediction distributions of unperturbed and perturbed samples.

4.3 Additional Design Choices

Prompting the feature extractor. Prompt tuning can be performed in a multitude of ways by changing the definition of $f(\theta_p, x)$. In our experiments, we observe that for adversarial robustness, using PT2 (otherwise known as Prefix Tuning) significantly outperforms using only PT. This behavior is consistently observed when using the same training scheme and total number of tuned parameters across the two methods. We hypothesize that this performance gap exists since the feature extractor Φ^* of the frozen backbone is not adversarially robust. Since PT simply prepends additional tokens only to the input of the frozen feature extractor, this results in the same function Φ^* on a different set of inputs. However, PT2 prepends prompt tokens directly in each layer of the feature extractor, resulting in a modified feature extractor $\hat{\Phi}$.

Tuning the embedding layer. Traditionally, the prompt tokens are introduced *after* the embedding function T and do not influence the mapping of the image patches to the embedding space. That is, the patch embedding remains unchanged by prompt tuning, which can cause vulnerability. Since the patch embedding does not have a significantly large number of parameters compared to the frozen backbone, we opt to tune the patch embedding parameters T along with the prompt tokens θ_p and the linear classifier head Ψ.

Algorithm 1 shows a pseudocode of our training algorithm. We provide our code as supplementary material, and we will release an open-source version of our code upon acceptance.

5 Experiments

We evaluate our methods on benchmark datasets with a focus on increasing adversarial robustness while keeping the number of tuned parameters relatively low. When using prompt tuning, we demonstrate the gradient obfuscation occurring in previous methods, and show that our proposed method overcomes this issue. Finally, we perform ablation studies on the different components of our method. To the best of our knowledge, we are the first to explore the robustness of the prompt tuning paradigm for ViTs.

It is imperative to note that the advantage of the prompt tuning paradigm is the adaptation of a single frozen backbone to different tasks by using a small number of additional parameters per task. For example,

Algorithm 1 ADAPT training step

Require: data pair x, y, frozen classifier f, tunable prompts θ_p, attack step size α, maximum perturbation magnitude ϵ, number of attack steps s, learning rate η

$x'_p \leftarrow x + U(-\epsilon, \epsilon)$ ▷ U is the uniform noise distribution

for $i = 1 \rightarrow s$ **do**

$x'_p \leftarrow x'_p + \alpha \nabla_{x'_p} \mathcal{L}_{CE}(f(\theta_p, x'_p), y)$ ▷ eq. (13)

$x'_p \leftarrow \text{Project}(x'_p, x - \epsilon, x + \epsilon)$ ▷ Project x'_p onto the ϵ ball

end for

$\mathcal{L} \leftarrow \mathcal{L}_{ADAPT}$ ▷ eq. (14)

$\theta_p \leftarrow \theta_p + \eta \nabla_{\theta_p} \mathcal{L}$ ▷ Optimizer gradient step

if a model was required for a hundred different downstream tasks, traditionally we would have to save and load a hundred fully fine-tuned copies of the large backbone with hundreds of millions of parameters for each task. Instead, prompt tuning requires training only a fraction of the original parameters per task to adapt a single frozen pre-trained model for each downstream task. With that said, prompt tuning does not benefit the time required for training or inference for the downstream tasks as the prompts are used in conjunction with the entire frozen backbone. This is true for any method that adopts the prompt tuning paradigm and not just our scenario. In any case, we report training time details in the appendix.

5.1 Experimental Setting

Datasets. We perform experiments on CIFAR-10(Krizhevsky et al., a), CIFAR100 (Krizhevsky et al., b), and Imagenette (Howard). Note that these datasets are widely employed benchmarks for evaluating adversarial defenses since training and evaluation on ImageNet using multi-step gradient-based attacks requires significant computational time and resources.

Adversarial Robustness Methods. We compare ADAPT to the following state-of-the-art methods established in the literature:

- *Adversarial Training* (Madry et al., 2017): As described in eq. (12), Adversarial Training solves a minimax optimization objective by augmenting the inputs during training using traditional adversarial attacks.

- *TRADES* (Zhang et al., 2019): TRADES identifies a trade-off between robustness and accuracy. It designs a defense method accordingly, which performs training with a modified loss function, performing natural training and at the same time reducing the prediction distance of unperturbed samples with that of samples perturbed to maximize the same distance.

Architectures and Tuned Parameters. We perform most of our experiments on the ViT-B model. For prompt tuning (PT), we keep the number of tuned parameters within $\sim 1\%$ of the total model parameters. We provide additional results on the ViT-S and ViT-L model configurations.

As discussed in design choices, due to superior performance, we perform most experiments with PT2 and tune the patch embedding as well as the linear classifier (all of which we take into account into the number of tuned parameters). We also perform ablation studies on PT while freezing the patch embedding layer.

Evaluation Metrics. We evaluate each method with regards to accuracy on the test set. We evaluate on unperturbed samples, also known as clean accuracy, as well as perturbed samples. In one experiment, as mentioned, we use traditional adversarial attacks as well as adaptive ones to validate our claims of gradient obfuscation. In all other sections, excluding the black-box attacks section, our attacks are white-box and adaptive to ensure a fair worst-case evaluation. We use adaptive PGD10 eq. (13), adaptive CW (Carlini & Wagner, 2017), which is obtained by replacing the loss in eq. (13) with the CW loss, as well as AutoAttack (AA) (Croce & Hein, 2020), an ensemble of diverse and parameter free white-box and black-box attacks.

Table 1: Test set accuracy of previous adversarial robustness methods combined with prompt tuning against unperturbed inputs, traditional adversarial attacks, PGD10 (multi-step) and FGSM (single step) as well as our adaptive attack. All methods perform worse against FGSM than PGD10, a clear sign of gradient obfuscation. Additionally, all methods show very little robustness to the adaptive attack (Adaptive PGD10). † indicates that the attack in question is non-adaptive while * indicates an adaptive attack eq. (13).

Params	Method	Clean	PGD†	FGSM†	PGD*
PT	AT	93.21	87.62	78.92	1.15
PT2 + Emb	AT	94.84	90.3	82.49	3.04
PT	TRADES	89.79	84.87	79.65	1.01
PT2 + Emb	TRADES	92.64	89.65	85.93	3.17

Finally, we evaluate against two state of the art blackbox attacks: RayS (Chen & Gu, 2020) and AdaEA (Chen et al., 2023b).

5.2 False Sense of Security

In this section, we evaluate the robustness of existing methods, combined with prompt tuning, against traditional and adaptive attacks. The results are presented in table 1 and are consistent with what we discussed in adaptive attacks section. The results show that using traditional adversarial examples and adversarial training can lead to gradient obfuscation (Athalye et al., 2018), and a false sense of security. Indeed, existing adversarial defense methods initially exhibit high performance when used with prompt tuning. However, as per (Athalye et al., 2018), a sign of gradient obfuscation is that single-step gradient-based attacks are more effective than their multi-step counterparts. As expected, previous methods are more susceptible to single-step FGSM attacks than multi-step PGD attacks and show significant vulnerability to our adaptive attack.

5.3 Adaptive Adversarial Training

We experiment with prompt tuning using previous SOTA adversarial defense methods and compare them to ADAPT. We present the results in table 2. The results show that while previous methods show little to no robustness against adaptive adversarial attacks, ADAPT achieves competitive accuracy on unperturbed samples while showing considerable robustness to adaptive adversarial attacks. We provide additional results for different model sizes, ViT-L and ViT-S, in the appendix, the results are consistent with that for ViT-B.

We also evaluate our methods on Imagenette (Howard), which is a subset of ImageNet with 10 classes, and CIFAR100 (Krizhevsky et al., b), which consists of 100 classes of 32×32 images. Results are presented in table 3 and table 4 respectively, and show consistency with the CIFAR10 experiments.

Note that for a fair comparison, we perform PT2 with the same number of tokens and tune the patch embedding and the linear classifier head for all methods, resulting in exactly the same number of tuned parameters. We include the results of full model fine-tuning (FT) as an upper bound on performance, which requires tuning all the ViT parameters. We conduct experimentation with PT and the effects of training the patch embedding in the ablation study (table 7). The results confirm the effectiveness of our design choices. Furthermore, we investigate the impact of the number of tuned parameters in the following sections.

5.4 Evaluation against Blackbox Attacks

In this section, we evaluate the robustness of ADAPT and existing methods against black-box attacks in the prompt-tuning paradigm. We evaluate against two SOTA black-box attacks: one query-based attack, RayS (Chen & Gu, 2020) and one transfer-based ensemble attack, AdaEA (Chen et al., 2023b). Table 5 shows that ADAPT is robust to **both** the query-based and transfer-based attacks, while AT and TRADES (with prompt tuning) only show robustness to the transfer-based attack and are **not robust** to the query-based attack. We hypothesize that the reason is AT and TRADES (with prompt tuning) are trained on attacks generated using the gradient of the naturally trained frozen ViT. The transfer-based attack also relies on the

Table 2: Test set accuracy on the CIFAR10 dataset for different methods. The upper section corresponds to methods using prompt tuning, and the lower section corresponds to FT as an upper bound on performance. Existing methods show little to no robustness to adaptive attacks. In the prompt tuning scenario, our method shows substantial improvement in robustness compared to previous methods. * indicates that the attack in question is adaptive (eq. (13)).

Method	Params.	Clean	PGD10*	CW*	AA	# params
AT	PT2+Emb	**94.84**	3.04	0.67	1.7	820K
TRADES	PT2+Emb	92.64	3.17	1.85	4.2	820K
ADAPT$_{CE}$	PT2+Emb	79.05	38.27	**35.94**	**19.90**	820K
ADAPT$_{KL}$	PT2+Emb	68.43	**40.21**	35.9	18.05	820K
AT	FT	82.42	53.41	46.67	34.77	86M
TRADES	FT	**84.02**	**54.19**	**47.45**	**36.48**	86M

Table 3: Test set accuracy on Imagenette

Method	Params	Clean	PGD10	CW	AA	# params
TRADES	PT2+Emb	**89.32**	9.04	6.82	7.08	820K
ADAPT$_{CE}$	PT2+Emb	59.95	31.34	28.25	**13.81**	820K
ADAPT$_{KL}$	PT2+Emb	63.16	**37.61**	**34.01**	10.96	820K
TRADES	FT	**82.4**	**53.4**	**43.6**	**16.33**	86M

Table 4: Test set accuracy on CIFAR100

Method	Params	Clean	PGD10	CW	AA	# params
TRADES	PT2+Emb	**78.84**	5.54	3.52	3.4	820K
ADAPT$_{CE}$	PT2+Emb	60.2	**22.15**	**19.59**	**7.92**	820K
ADAPT$_{KL}$	PT2+Emb	50.63	21.77	17.56	5.94	820K
TRADES	FT	**65.5**	**32.89**	**28.93**	**9.99**	86M

gradients of naturally pre-trained models to construct its attack, while the query-based attack eliminates the need to use gradients. This aligns with our discussion of gradient obfuscation. Furthermore, against existing defenses, both black-box attacks are significantly more successful than white-box non-adaptive attacks (see table 2), which is another sign of gradient obfuscation in existing defenses when using prompt tuning.

Table 5: Test-set accuracy against black-box attacks, same training settings as Table 2.

Method	Params	RayS	AdaEA
AT	P2T+Emb	5.06	66.55
TRADES	P2T+Emb	11.71	66.27
ADAPT$_{CE}$	P2T+Emb	43.05	58.21
ADAPT$_{KL}$	P2T+Emb	40.09	52.06
TRADES	FT	56.78	61.73

5.5 Beyond the Number of Parameters

We notice in our experiments that for adversarial training under the same conditions, PT2 significantly outperforms PT. We hypothesized that this is due to the fact that PT is simply augmenting the input to the frozen feature extractor $\Phi*$ of the transformer which is vulnerable to adversarial examples, while PT2 inserts tokens in the intermediate layers of the feature extractor, modifying it directly. We perform an experiment to validate our intuition and confirm that the superior performance of PT2 over PT, is not simply due to

an increased number of parameters (as with a fixed number of tokens per layer, PT2 requires more tuned parameters).

We consider a set of experiments with a fixed number of tuned tokens, and we gradually change the scenario from PT to PT2 in discrete steps as described below. We tune a fixed number of M tokens, prompting the first k layers of the transformer with m tokens each, with varying numbers of k and m while keeping $m \times k = M$.

For example, we consider a pool of 96 tokens. In the first experiment, we perform training and testing while only prompting the first layer with all 96 tokens (which is equivalent to PT). In the following experiments, we perform training and testing while prepending to more and more layers, e.g. the first two layers with 48 tokens, and so on, until prepending to all 12 layers each with 8 tokens (which is equivalent to PT2). In all of the above experiments, the same number of tuned parameters is used (96 tokens, plus the classifier head).

The results are presented in table 6, and show a substantial increase in performance as we prompt more and more layers despite using the same number of tokens. This further validates our insight about needing to modify the feature extractor and shows that the performance improvement is not simply due to an increase in the number of tuned parameters.

Table 6: Test set accuracy and the number of tuned parameters of different combinations. $m \times k$ means we prompt the first k layers directly with m tokens each. Each setting is adopted during both training and testing. We measure against unperturbed samples and the adaptive PGD10 attack.

Combination	Clean	PGD10	# params
96×1	27.33	18.34	81K
48×2	32.74	21.07	81K
24×4	40.03	24.83	81K
16×6	40.78	25.49	81K
12×8	**41.14**	**25.68**	81K
8×12	40.24	25.17	81K

Table 7: Ablation study on different prompt tuning scenarios on the CIFAR10 dataset. Performance is measured by test set accuracy againt unperturbed samples and samples perturbed by adaptive PGD10.

Method	Params	Clean	PGD10	# params
ADAPT$_{CE}$	PT	44.72	15.24	19K
ADAPT$_{KL}$	PT	39.77	9.54	19K
ADAPT$_{CE}$	PT2	67.45	26.75	230K
ADAPT$_{KL}$	PT2	58.4	29.25	230K
ADAPT$_{CE}$	PT+Emb	67.8	29.44	600K
ADAPT$_{KL}$	PT+Emb	56.48	30.31	600K
ADAPT$_{CE}$	PT2+Emb	**79.05**	38.27	820K
ADAPT$_{KL}$	PT2+Emb	67.24	**39.23**	820K

6 Conclusion

We formulated and extensively analyzed the adversarial robustness of ViTs while performing parameter-efficient prompt tuning. We found that the application of existing adversarial defense methods to the prompt tuning paradigm suffers from gradient obfuscation. This leads to a *false sense of security* and a vulnerability to adaptive attacks, which we demonstrated through experiments with single-step and adaptive attacks. To address these problems, we proposed a novel loss function for adaptive adversarial training of prompt tuning methods. We evaluated our method on various datasets and model sizes, and performed an ablation study on the components of our framework. We empirically demonstrated the superior adversarial robustness of our method while performing parameter-efficient prompt tuning.

7 Broader Impact Statement

Adversarial perturbations can be used to fool machine learning models to make decisions based on the adversary's intent which can be malicious. For example, in facial recognition for security systems, an adversary may try to pose as someone else to breach the system. In self-driving cars, an adversary can change the classification of traffic signs or objects to alter the course of the car and cause deadly crashes. As such, it is imperative to explore and improve the robustness of machine learning models. In this work, we showcase the significant vulnerability of previous adversarial defense methods to adaptive adversarial attacks under the prompt tuning paradigm. To this end, we provide a framework for adaptive adversarial training of prompts to combat vulnerability to adaptive attacks. This makes models using the prompt tuning paradigm more robust to adversarial manipulation.

References

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pp. 274–283. PMLR, 2018.

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

Jacob Buckman, Aurko Roy, Colin Raffel, and Ian Goodfellow. Thermometer encoding: One hot way to resist adversarial examples. In *International conference on learning representations*, 2018.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pp. 39–57. Ieee, 2017.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

Aochuan Chen, Peter Lorenz, Yuguang Yao, Pin-Yu Chen, and Sijia Liu. Visual prompting for adversarial robustness. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2023a.

Bin Chen, Jiali Yin, Shukai Chen, Bohao Chen, and Ximeng Liu. An adaptive model ensemble adversarial attack for boosting adversarial transferability. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4489–4498, 2023b.

Jinghui Chen and Quanquan Gu. Rays: A ray searching method for hard-label adversarial attack. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1739–1747, 2020.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pp. 2206–2216. PMLR, 2020.

Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. In *International Conference on Machine Learning*, pp. 7480–7512. PMLR, 2023.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Gamaleldin F Elsayed, Ian Goodfellow, and Jascha Sohl-Dickstein. Adversarial reprogramming of neural networks. *arXiv preprint arXiv:1806.11146*, 2018.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.

Jeremy Howard. Imagewang. URL `https://github.com/fastai/imagenette/`.

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

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pp. 709–727. Springer, 2022.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pp. 4171–4186, 2019.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). a. URL `http://www.cs.toronto.edu/~kriz/cifar.html`.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-100 (canadian institute for advanced research). b. URL `http://www.cs.toronto.edu/~kriz/cifar.html`.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3045–3059. ACL, November 2021. doi: 10.18653/v1/2021.emnlp-main.243. URL `https://aclanthology.org/2021.emnlp-main.243`.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35, 2023.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Lam Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *arXiv preprint arXiv:2110.07602*, 2021.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Kaleel Mahmood, Rigel Mahmood, and Marten Van Dijk. On the robustness of vision transformers to adversarial examples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7838–7847, 2021.

Yichuan Mo, Dongxian Wu, Yifei Wang, Yiwen Guo, and Yisen Wang. When adversarial training meets vision transformers: Recipes from training to architecture. *Advances in Neural Information Processing Systems*, 35:18599–18611, 2022.

OpenAI. Gpt-4 technical report, 2023.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. Training data-efficient image transformers; distillation through attention. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 10347–10357. PMLR, 18–24 Jul 2021. URL `https://proceedings.mlr.press/v139/touvron21a.html`.

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

Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 139–149, 2022.

Ying Xu, Xu Zhong, Antonio Jimeno Yepes, and Jey Han Lau. Grey-box adversarial attack and defence for sentiment classification. *arXiv preprint arXiv:2103.11576*, 2021.

Yuting Yang, Pei Huang, Juan Cao, Jintao Li, Yun Lin, Jin Song Dong, Feifei Ma, and Jian Zhang. A prompting-based approach for adversarial example generation and robustness enhancement. *arXiv e-prints*, pp. arXiv–2203, 2022.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pp. 7472–7482. PMLR, 2019.

Jingfeng Zhang, Xilie Xu, Bo Han, Gang Niu, Lizhen Cui, Masashi Sugiyama, and Mohan Kankanhalli. Attacks which do not kill training make adversarial learning stronger. In *International conference on machine learning*, pp. 11278–11287. PMLR, 2020.

Zangwei Zheng, Xiangyu Yue, Kai Wang, and Yang You. Prompt vision transformer for domain generalization. *arXiv e-prints*, pp. arXiv–2208, 2022.

A Different Model Sizes

We provide results for the experiment setting of table 1, for different ViT configurations, namely ViT-S in table 8 and ViT-L in table 9. The results are consistent with that of table 1.

Table 8: CIFAR10 test set accuracy for the ViT/Small model

Method	Params.	Clean	PGD	CW	# params
TRADES	PT2+Emb	**86.06**	3.98	3.81	740K
ADAPT+CE	PT2 + Emb	73.7	34.33	31.52	740K
ADAPT+KL	PT2 + Emb	63.37	**36.1**	**31.74**	740K
TRADES	FT	**81.1**	**51.71**	**45.16**	57M

Table 9: CIFAR10 test set accuracy for the ViT/Large model.

Method	Params.	Clean	PGD	CW	# params
TRADES	PT2+Emb	**94.12**	3.68	2.36	1.4M
ADAPT+CE	PT2+Emb	80.79	39.49	**36.82**	1.4M
ADAPT+KL	PT2+Emb	69.98	**41.16**	**36.82**	1.4M
TRADES	FT	**83.88**	**54.67**	**48.12**	307M

B Training Time Analysis

Prompt tuning seeks to adapt large models to downstream tasks with minimal parameters. This allows us to load and store only one set of weights for a large backbone, and that set can be adapted by storing and loading a low number of parameters for each downstream task.

Subsequently, it is important to note that the advantage of prompt tuning is a reduction in the number of tuned parameters and not a reduction in training time, as the forward and backward passes still flow through

the entire model. This results in a slight increase in computation cost, as adding prompt tokens increases the number of tokens that go through the transformer. *This is the case for any application of PT and PT2 and is* ***not*** *specific to our scenario.* However, in our experiments, we empirically observe that while tuning the prompts, we can train for fewer epochs with a cyclic learning rate to achieve our best-case performance. The same was not observed in the fine-tuning scenario and more training epochs were required to achieve the best performance in terms of validation accuracy.

With all that said, we provide training time details for the scenarios in table 10 for those interested. We report the training time for each method in minutes. Each method was trained using an NVIDIA Tesla V100 SXM2.

Table 10: Total training time as measured in hours (rounded to the closest integer) for the methods presented in Tab. 2 of the main paper. Each method in the prompt tuning scenario was trained for 20 epochs with a cyclic learning rate while each method in the fine tuning scenario was trained for 40 epochs with an annealing learning rate.

Method	Params.	Total Training Time (hours)
AT	PT2+Emb	15
TRADES	PT2+Emb	16
ADAPT-CE (Ours)	PT2+Emb	18
ADAPT-KL (Ours)	PT2+Emb	18
AT	FT	11
TRADES	FT	13