Revisiting Adversarial Training at Scale

Zeyu Wang* Xianhang Li* Hongru Zhu Cihang Xie

*equal contribution

UC Santa Cruz

Abstract

The machine learning community has witnessed a drastic change in the training pipeline, pivoted by those "foundation models" with unprecedented scales. However, the field of adversarial training is lagging behind, predominantly centered around small model sizes like ResNet-50, and tiny and low-resolution datasets like CIFAR-10. To bridge this transformation gap, this paper provides a modern re-examination with adversarial training, investigating its potential benefits when applied at scale. Additionally, we introduce an efficient and effective training strategy to enable adversarial training with giant models and web-scale data at an affordable computing cost. We denote this newly introduced framework as AdvXL.

*Empirical results demonstrate that AdvXL establishes new state-of-the-art robust accuracy records under AutoAttack on ImageNet-1K. For example, by training on DataComp-1B dataset, our AdvXL empowers a vanilla ViT-g model to substantially surpass the previous records of l_∞-, l_2-, and l_1-robust accuracy by margins of **11.4%**, **14.2%** and **12.9%**, respectively. This achievement posits AdvXL as a pioneering approach, charting a new trajectory for the efficient training of robust visual representations at significantly larger scales. Our code is available at https://github.com/UCSC-VLAA/AdvXL.*

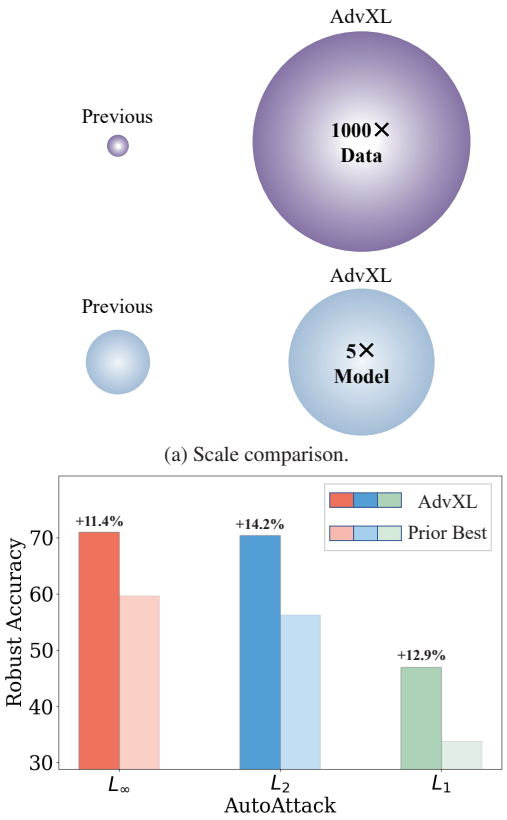


Figure 1. Our AdvXL increases significantly in terms of both model size and data scale, which brings a substantial boost over prior best results of l_∞, l_2, and l_1 robustness on ImageNet-1K, even though our model is only trained to be l_∞-robust.

1. Introduction

The landscape of machine learning, particularly deep learning, has witnessed a transformative shift with the advent of large-scale models and datasets. This paradigmatic shift, exemplified by the inception of "foundation models" such as Large Language Models (LLMs) [6, 14, 41, 55, 56], has redefined the boundaries of what is achievable in various domains of artificial intelligence. Excitingly, parallel developments have also been observed in computer vision — recent advancements in scaling datasets and model sizes have mirrored the feasibility of "LLM-like" scaling for building exceptionally strong visual recognition models [12, 16, 64].

However, amidst this evolution, adversarial training [19, 39] — a pivotal strategy aimed at securing model robustness against adversarial attacks — has faced significant scalability challenges in this foundation model era. Adversarial training, typically employed in small models such as ResNet-50 [23] trained on small datasets like CIFAR-10 [28], involves repeatedly generating adversarial examples through on-the-fly attacks during the training process. This iterative and intensive procedure demands substantial computational resources, thus making it challenging to scale up.

Contrasting with these challenges, recent endeavors in adversarial training have indeed shown intriguing glimpses of promise from data scaling by incorporating 50 million additional images to sustain state-of-the-art robustness records on CIFAR-10 [57]. Additionally, other adversarial training works [34, 52] attain impressive performance with model scaling using larger models like Swin-L [35] and ConvNeXt-L [37] on ImageNet-1K. These observations, coupled with the burgeoning success of foundation models, instigates a critical question: *can the principles of model and data scaling, already proven effective in vanilla training, be transferable to adversarial training?* Moreover, how effectively does such scaling translate to robustness improvement in adversarial training?

In response to these questions, we re-examine adversarial training at a previously uncharted foundation-model scale. In terms of model scaling, we increased the model parameters from the previously largest 200M size to **1B**; for data scaling, we adversarially train models on various datasets spanning from the medium-size ImageNet-1K with around 1M images to the web-scale dataset comprising more than **1B** images. Additionally, to make the scaling of adversarial training computationally affordable, we introduce an efficient approach with a straightforward two-stage training schedule, *i.e.*, first lightweight pre-training, then intensive fine-tuning. We name this efficient and scalable adversarial training framework as AdvXL.

Collectively, extensive experiments showcase that these scaling endeavors successfully result in substantial improvements over the previous state-of-the-art methods on adversarial robustness. For example, by training a one-billion parameter model on a one-billion image dataset, we establish a new state-of-the-art record for l_∞-robust accuracy of 71.0% under AutoAttack on ImageNet-1K, marking a substantial enhancement in model robustness. Notably, AdvXL demonstrates exceptional generalizability when tested against unseen attacks, improving upon the previous best l_2- and l_1-robust accuracy of models trained to be l_∞-robust by margins of $\sim$14% and $\sim$13%, respectively. These results underscore the pivotal role of (significantly) scaled adversarial training in enhancing model robustness against diverse adversarial threats.

2. Related Work

2.1. Adversarial Training

Adversarial training has emerged as a pivotal defense mechanism against adversarial attacks in machine learning. Initially introduced by Goodfellow *et al*. [19], this methodology involves training models on crafted adversarial examples designed to provoke model misclassification. Subsequent studies have extended this foundation, examining facets such as the impact of batch size, learning rate, data augmentation, and training duration on model robustness, predominantly on smaller datasets like CIFAR-10 [7, 20, 25, 33, 40, 42]. Other research efforts have explored deeper nuances of adversarial training recipes tailored for ImageNet-1K [3, 11, 49, 52, 60–62]. Recent works also investigate the robustness of novel network designs like Vision Transformer (ViT) [9, 17, 21, 52]. In particular, Singh *et al*. [52] achieve the best generalized robustness by enhancing ViT and ConvNeXT with Convolutional Stem.

Despite its effectiveness, adversarial training is notoriously resource-intensive, limiting its scalability. To address this challenge, researchers have pursued more resource-efficient adversarial training methodologies. Examples include Free Adversarial Training [51] and Fast Adversarial Training [58], both aimed at reducing training costs while preserving model robustness. However, these approaches have predominantly focused on smaller networks and datasets, leaving a noticeable gap concerning large-scale models. In this work, we aim to significantly expand the horizons of scaling adversarial training to unprecedented levels of efficiency and effectiveness.

2.2. Scaling Vision Foundation Models

Parallel to large-scale language models, exemplified by innovations like GPT series [41], similar efforts have been made for vision models, particularly with the scaling of ViTs [12, 16, 64]. Liu *et al*. [36] effectively trained the SwinV2-G model, housing an astounding 3B parameters, by employing residual-post-norm and scaled cosine attention. Similarly, Dehghani *et al*. [12] have shown substantial performance enhancements by scaling ViTs to 22B parameters, mirroring the scaling trends witnessed in language models.

Despite the burgeoning scaling efforts in vision foundation models, the exploration of adversarial training has traditionally been limited to small or base model sizes. Recent scaling effort has led to noteworthy performance improvements, evidenced by the achievements on RobustBench [9] with larger models like Swin-L and ConvNeXt-L [34, 52]. Diverging from these antecedent initiatives, our work explores adversarial training at an even much larger scale, up to the training of a one-billion-parameter model on one-billion samples, thereby pioneering the frontiers of adversarial training into uncharted territory.

3. AdvXL

In this section, we introduce AdvXL, a novel training framework designed for adversarially robust visual representation learning at scale. We first revisit the fundamental concept of adversarial attacks and adversarial training in Sec. 3.1. Following this, in Sec. 3.2, we present a two-stage efficient adversarial training pipeline characterized by a coarse-to-fine, weak-to-strong approach. In Sec. 3.3, we

showcase how to leverage CLIP [47] text encoder as a tool for enabling us to learn with web-crawled images, where a precise label is usually missing but with a corresponding text description, for scaled adversarial training.

3.1. Adversarial Training

Adversarial examples are uniquely crafted inputs that, despite their visual similarity to authentic samples within specific norm constraints, are engineered to deceive machine learning models into producing inaccurate predictions. These examples play a crucial role in assessing the robustness of a model in scenarios where malicious manipulations may occur.

Adversarial Training is central to fortifying a model against such adversarial inputs. This technique involves a strategic training process designed to enhance the model's robustness to adversarial attacks. The mathematical foundation of AT is encapsulated as an optimization problem:

$$\min_{\theta} \sum_{(x_i,y_i)\in\mathcal{D}} \max_{\delta:\|\delta\|_p\leq\epsilon_p} \mathcal{L}\left(f_\theta\left(x_i+\delta\right), y_i\right), \qquad (1)$$

where θ represents the parameters for a network f_θ. The objective is to train the network f_θ such that it maintains consistent predictions under adversarial perturbations δ, *i.e.*, within an l_p-ball of radius ϵ_p centered around each input sample x_i.

Adversarial Training has proven highly effective to safeguard models against adversarial threats [5, 19, 53]. In our approach, we adopt the widely recognized PGD-based Adversarial Training (PGD-AT) method for the inner maximization problem, renowned for its robust performance and computational efficiency. For the outer minimization problem, we typically employ optimization algorithms like Stochastic Gradient Descent or AdamW [38], using cross-entropy as the loss function $\mathcal{L}$.

3.2. Two-stage Training

Our adversarial training framework hinges on a two-stage process: a lightweight pre-training stage and an intensive fine-tuning stage. During the pre-training stage, the model is trained with inputs at reduced token length and weaker attacks, spanning a relatively extended duration. Then, during the subsequent fine-tuning stage, the model is trained with inputs at full resolution and stronger attacks, following a comparatively shorter schedule. Compared to the vanilla one-stage adversarial training pipeline, this coarse-to-fine (*w.r.t.* input), weak-to-strong (*w.r.t.* adversarial attacker), two-stage training pipeline significantly reduces the overall training cost, rendering it computationally affordable for further scaling up.

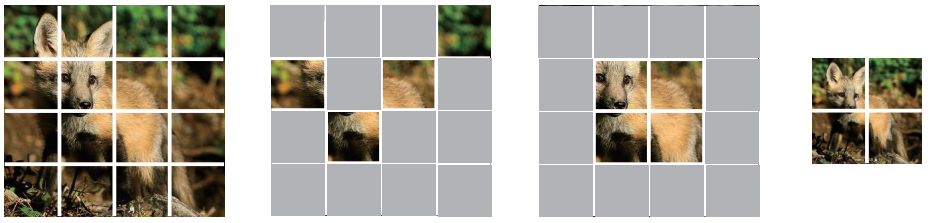
Coarse-to-fine training. We first explore various strategies for image token reduction in the initial pre-training stage. Following [30, 31], three distinct approaches are investigated:

- *Random Masking*. This method, as described in [24, 32], involves dividing an image into non-overlapping patches (*e.g.*, 16×16), subsequently masking a random proportion of these patches (*e.g.*, 75%). The model only processes the visible patches, reducing the computational cost by 50% or 75%, depending on the masking ratio.
- *Block Masking*. Inspired by [4], this approach retains tokens from a consecutive large block within the image while discarding others. This method leverages the common placement of objects in the central regions of images, potentially preserving semantic meaningful tokens while significantly reducing the computational cost from lengthy inputs.
- *Resizing*. Image resizing is another method for reducing the image token length. Compared to masking, resizing retains more image information, especially high-level semantics. For instance, resizing an image to 112×112 is computationally akin to applying a 75% masking ratio to an image resized to 224×224. In our approach, we choose anti-aliasing bilinear interpolation to better preserve the image quality.

A visual comparison illustrating these image token reduction strategies is presented in Fig. 2. These strategies are evaluated to discern their efficacy in achieving training acceleration while retaining critical image semantics.

Weak-to-strong training. Another critical factor for accelerating adversarial training involves managing the number of gradient steps used to craft adversarial samples. Generally speaking, increasing the number of gradient steps results in stronger attacks and enhances adversarial robustness but inevitably inflates computational costs. It has been reported that forming a robust network with adversarial training can take significantly longer, ranging from 3 to 30 times more than building a non-robust equivalent [51]. As a result, previous studies [44, 51, 59] have proposed strategies like recycling gradient information or employing a small generator network to mitigate the significant computational burden in adversarial training.

Our exploration reveals that applying a small number of PGD steps (*e.g.*, PGD-1) during the pre-training stage and subsequently increasing these steps during the fine-tuning phase (*e.g.*, PGD-3) sufficiently secure strong robustness, *i.e.*, this method proves effective compared to initiating training with strong attacks. Importantly, this approach contributes a notable additional speedup, enhancing the efficiency gained from the coarse-to-fine training pipeline (*e.g.*, up to 2×), as solving the inner optimization of adversarial training often requires optimization with multiple iterations and is extremely time-consuming.



(a) Original (b) Random Masking (c) Block Masking (d) Resizing

Figure 2. Illustration of different approaches to image token reduction.

Fine-tuning. Echoing findings from prior research [31, 32], we find that further adversarially training our model with full-resolution inputs and stronger attacks for a short schedule yields considerable improvement and delivers a more favorable accuracy-to-time trade-off. Compared to the pre-training stage, the fine-tuning phase is notably shorter, often reduced by one or two orders of magnitude. Therefore, even though each sample may entail a notably higher number of image tokens (*e.g.*, $4\times$ by switching back to full image resolution) and require more gradient steps (*e.g.*, $2\times$ by switching back to the strong PGD-3 attacker) in this fine-tuning phase, the overall computation does not increase significantly.

3.3. CLIP Embedding for Web-Crawled Images

Previous works have leveraged the zero-shot generalization capability of pre-trained CLIP text encoder [47] to aid a range of downstream tasks, including object detection [22, 66, 67] and segmentation [29, 48] in an open-vocabulary setting. Similarly, we hereby propose to employ CLIP text encoder to extract classifier weights when training on web-crawled large-scale datasets with open text descriptions, such as LAION-400M [50] and DataComp-1B [18]. Moreover, adversarial training on these gigantic datasets enables the model to transcend pre-defined categories and directly learn intricate class relationships through natural language supervision.

Specifically, we adopt the contrastive loss from [47, 54], formulated as:

$$\mathcal{L}\left(f^I, f^T, I_i, T_i\right) = -\frac{1}{2n}\sum_i\left(\log\frac{\exp\left(h_i^T\cdot h_i^I/\tau\right)}{\sum_j\exp\left(h_i^T\cdot h_j^I/\tau\right)}+\log\frac{\exp\left(h_i^I\cdot h_i^T/\tau\right)}{\sum_j\exp\left(h_i^I\cdot h_j^T/\tau\right)}\right) \tag{2}$$

where n represents the batch size; τ is a learnable temperature parameter; $h_i^I = f^I\left(I_i\right)/\left|f^I\left(I_i\right)\right|$ and $h_i^T = f^T\left(T_i\right)/\left|f^T\left(T_i\right)\right|$ denote the normalized projected features of an image-text pair (I_i, T_i). Note that we opt for CLIPA-trained text encoder [30, 31] as the initial f^T weight and keep it frozen during training. In this case, the adversarial training framework can be described as the following optimization problem,

$$\min_{\theta^I}\sum_{(I_i,T_i)\in\mathcal{D}}\max_{\delta:\|\delta\|_p\le\epsilon_p}\mathcal{L}\left(f^I, f^T, I_i+\delta, T_i\right), \tag{3}$$

where θ^I represents the parameters of the image encoder f^I. To elucidate this integration further, Fig. 3 provides a visual representation illustrating the incorporation of the CLIP encoder in adversarial training.

4. Experiment

In this section, we first introduce the datasets used for adversarial training, along with the details of the training and evaluation setup in Sec. 4.1. In Sec. 4.2, we delve into the ablation results, exploring key elements in our two-stage training pipeline. Furthermore, we investigate the performance of adversarial training as the model, data, and schedule scale synergistically in Sec. 4.3. Finally, we compare and contrast the efficiency and efficacy of AdvXL against prior arts in Sec. 4.5.

4.1. Implementation

Dataset. We utilize four different datasets as the training set for adversarial training, which are ImageNet-1K and ImageNet-21K [13] — two well-curated labeled datasets for supervised training, as well as LAION-400M [50] and DataComp-1B [18] — two weakly labeled datasets with natural language captions crawled from the Internet.

Specifically, ImageNet-1K comprises approximately 1.28M images from 1000 classes, while ImageNet-21K consists of around 13M images from 19k classes. LAION-400M is the first publicly available web-scale dataset consisting of 400M image-text pairs. It is filtered by CLIP and NSFW criterion, but is still relatively non-curated. DataComp-1B is a more recent dataset with about 1.3B samples filtered from a candidate pool of 12.8B image-text pairs from Common Crawl, which has been recorded to yield superior performance for contrastive training.

To summarize, our choices of training datasets cover a wide range of representative datasets, spanning from ~1M to ~1B samples, from well-curated labeled data to non-curated web data. This diverse selection enables a comprehensive investigation into the adversarial training concerning data scaling behaviors.

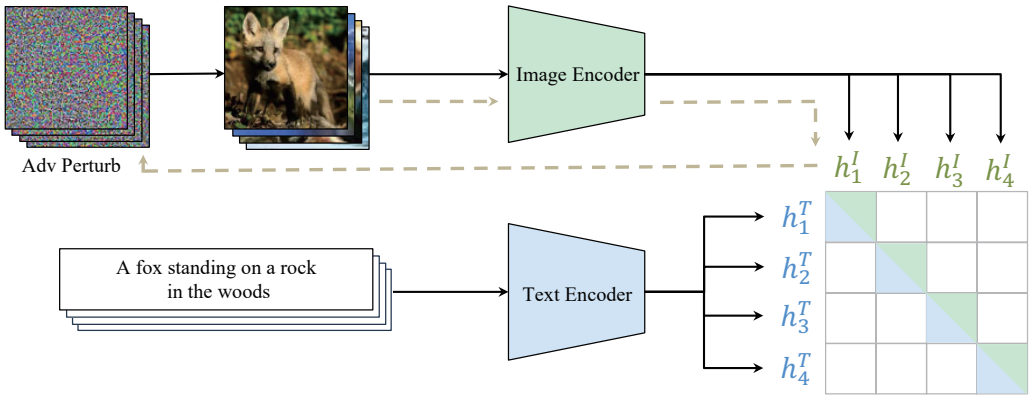


Figure 3. Illustration of leveraging CLIP embedding in adversarial training. The gray line denotes the adversarial example generation flow.

Training. By default, our training initiates with a pre-training stage utilizing an image size of 112 × 112 and PGD-1 with a step size of 4/255. Subsequently, the model undergoes a fine-tuning stage employing an image size of 224 × 224 and PGD-3 with a step size of 4/255. Our primary focus centers on ViT [15], renowned for its scalability [15, 24, 32, 47] yet relatively underexplored in the realm of adversarial training. Note that the current best ViT model on ImageNet-1K in RobustBench is only ViT-B/16 [9], indicating plenty of room for further scaling.

On ImageNet-1K and ImageNet-21K, our recipe closely follows prior works [24], which successfully trains ViTs on ImageNet at scale from scratch. Specifically, we adopt the AdamW optimizer [38] with a short-term linear learning rate warmup followed by a cosine learning rate schedule. Our data augmentation strategy integrates RandAug [10], MixUp [65] and CutMix [63]. Additionally, we incorporate stochastic depth [27] and weight decay for model regularization. On web-scale datasets such as LAION-400M and DataComp-1B, our training recipe aligns with methodologies outlined in [31].

The specifics of our training schedules are tailored to individual datasets, where the total number of training samples serves as the primary metric, following a paradigm akin to CLIP training [31, 32, 47]. For instance, our default pre-training schedule on ImageNet-1K spans a total of 256M samples, which corresponds to 200 epochs of training.

Evaluation. In our analysis, we primarily use robust accuracy under PGD-20 attack with a step size of 1/255 as the principal metric. When comparing against other state-of-the-art methods, we follow RobustBench [9] and use the robust accuracy evaluated on a subset of selected 5000 images of the ImageNet-1K validation set under AutoAttack. AutoAttack is a standardized adversarial robustness benchmark that consists of an ensemble of white- and black-box attacks, including APGD for cross-entropy and targeted DLR loss, FAB-attack [8] and the black-box Square Attack [2]. The attack radii are ϵ_∞ = 4/255, ϵ_2 = 2, and ϵ_1 = 75 for l_∞, l_2, and l_1 attacks, respectively.

Approach	Ratio/Size	Compute	Clean	PGD-20
baseline	224/0%	1.0×	75.5	54.5
Random Masking	224/50%	0.5×	72.0	51.9
Random Masking	224/75%	0.25×	67.3	46.5
Block Masking	224/50%	0.5×	72.3	52.0
Block Masking	224/75%	0.25×	70.6	49.3
Resizing	160/0%	0.5×	74.7	53.9
Resizing	112/0%	0.25×	73.0	52.5
Resizing	96/0%	0.18×	70.0	49.9

(a) **Image token reduction**.

Stage	Step	Step_size	Compute	Clean	PGD-20
Pre-training	1	4/255	1.0×	73.0	52.5
	2	3/255	1.5×	72.1	52.6
	3	3/255	2.0×	71.8	52.5
Fine-tuning	1	4/255	1.0×	75.0	50.6
	2	4/255	1.5×	73.2	52.3
	3	4/255	2.0×	73.0	52.5

(b) **Attack strength**.

Approach	Ratio/Size	Clean	PGD-20
w/o Tuning	160/0%	74.4	43.2
	112/0%	68.5	39.3
w Tuning	160/0%	74.7	53.9
	112/0%	73.0	52.5

(c) **Fine-tuning**.

Table 1. **Ablating design choices** with ViT-B/16 on ImageNet-1K. We report clean and PGD-20 robust accuracy (%). If not specified, the default setting is: *112×112 image size for pre-training, 224×224 image size for fine-tuning; PGD-1 for pre-training, and PGD-3 for fine-tuning; 200 epochs for pre-training length, 20 epochs for fine-tuning*. Default settings are marked in gray. In table (a) and (b), note that full-resolution fine-tuning is included. In table (b), when tuning the PGD step and step size in pre-training, we fix them to be 3 and 4/255 respectively in fine-tuning; When tuning the PGD step and step size in fine-tuning, we fix them to be 1 and 4/255 respectively in pre-training.

4.2. Design Choices

We first conduct an ablation study on the design choices of AdvXL using ViT-B/16 on ImageNet-1K, with robust accuracy under PGD-20 serving as the primary metric for adversarial robustness. We maintain the default baseline setting (see the caption of Tab. 1). Any alterations are confined to the specific factors under examination.

Token Reduction. Our investigation delves into three distinct strategies for reducing image token length: 1) random masking, which randomly removes a portion of input tokens; 2) block masking, which retains a large consecutive block of the input grid; 3) resizing, which preserves most high-level semantic information. As shown in Tab. 1a, all three methods exhibit substantial computational speedups. Notably, image resizing demonstrates superior performance among these strategies, presumably because it suffers the least from loss of information. For instance, resizing the input image to 112×112 leads to a 75% reduction in total computation, with only a minor decrease of 2.5% in clean accuracy and 2.0% in PGD-20 robust accuracy. *We select an image size of* 112×112 *for pre-training as the default setting due to its satisfactory balance between efficiency and performance*.

Attack Strength. Tab. 1b scrutinizes the impact of varying attack steps and step sizes during pre-training and fine-tuning. Intriguingly, we observe that the number of PGD steps for pre-training does not need to align with that for fine-tuning. For instance, adopting PGD-1 for pre-training yields nearly equivalent robustness compared to PGD-3, while reducing the computation by 100%. This suggests that despite exposure to weaker attacks during pre-training (*e.g.*, with PGD-1), a short-term but stronger adversarial fine-tuning (*e.g.*, with PGD-3) is sufficient for the model to secure strong robustness against adversarial attacks. Therefore, *we opt to use PGD-1 for pre-training and PGD-3 for fine-tuning in our default setting*.

Fine-tuning. Tab. 1c outlines the impact of full resolution fine-tuning with stronger attacks for an extra 20 epochs on the ImageNet-1K dataset. For 112×112 PGD-1 pre-training, a 224×224 PGD-3 fine-tuning elevates clean accuracy by 4.5% and PGD-20 robust accuracy by 13.2%. This fine-tuning phase substantially narrows the performance gap between reduced-length pre-training and full-length training, demanding only around 60% of the pre-training computational resources. Extending the pre-training schedule by the corresponding compute yields significantly inferior results, highlighting the distinct advantage of fine-tuning in achieving a superior performance-compute trade-off. Therefore, *we consistently integrate a short-term fine-tuning stage post pre-training*.

4.3. Scaling Behavior

The acceleration outlined previously allows us to delve into the performance implications of scaling AdvXL within an affordable computational budget. In particular, we scrutinize the scaling behavior along three principal axes below, in line with the approach established by Li *et al*. [32]:

- *Model scaling*. We substitute the ViT-B/16 model with ViT-L/16 or ViT-H/14, which has $\sim 2\times$ or $\sim 4\times$ number of parameters, respectively.
- *Data scaling*. We substitute the training set of ImageNet-1K with three much larger datasets, excessively expanding the total number of training samples up to more than $\sim$1B. These datasets include ImageNet-21K [13], a superset of ImageNet-1K; LAION-400M [50], and DataComp-1B [18], two web-scale datasets.
- *Schedule scaling*. To delineate the influence of large dataset size from that of extended training duration, we conduct training on ImageNet-21K with the same number of seen samples as training on ImageNet-1K.

By meticulously traversing these three scaling axes, we scrutinize their individual effects on AdvXL's performance. The findings are detailed in Tab. 2, culminating in the following insights.

Model scaling. The evaluation of larger model sizes reveals discernible improvements in both clean accuracy and adversarial robustness. For instance, as shown in the first and the second rows of Tab. 2, ViT-L/16 surpasses ViT-B/16 by 1.8% clean accuracy (from 73.0% to 74.8%) and 1.8% PGD-20-robust accuracy (from 52.5% to 54.7%) when training on ImageNet-1K. Interestingly, ViT-H/14, despite its superior clean accuracy and tripled computational expense, demonstrates only a slightly better performance (0.2% higher PGD-20 robustness) compared to ViT-L/16 when training on ImageNet-1K, as shown in the third row of Tab. 2. However, it notably surpasses ViT-L/16 by a substantial margin (2.2% in PGD-20 robustness) when training on the larger ImageNet-21K dataset (as shown in the fifth and sixth rows of Tab. 2). This observation suggests that larger models necessitate a larger training set to fully leverage their potential. This finding aligns with conclusions in prior studies [26], advocating for equivalent scaling of model size and the volume of training tokens.

Schedule scaling. Initial experiments demonstrated that extending the training schedule for ViT-L/16 on ImageNet-1K yielded diminishing gains, possibly due to the comparatively "limited" scale of ImageNet-1K. However, results in Tab. 2 shows that with larger and more diverse datasets, training with additional samples yields non-trivial enhancements. Even with a $20\times$ augmentation in the training schedule using a one-billion sample dataset, such as training a

Case	Model	Dataset	Samples@Resolution	Adv. Steps	Compute (1e10)	Clean	PGD-20
Baseline	ViT-B/16	ImageNet-1K	256M@112 + 38.4M@224	1/3	0.5	73.0	52.5
model scaling	**ViT-L/16**	ImageNet-1K	256M@112 + 38.4M@224	1/3	1.7	74.8	54.7
model scaling	**ViT-H/14**	ImageNet-1K	256M@112 + 38.4M@224	2/3	5.7	76.5	54.9
model+data scaling	**ViT-L/16**	**+ ImageNet-21K**	256M@112 + 38.4M@224	1/3	1.7	75.8	56.1
model+data+schedule scaling	**ViT-L/16**	**+ ImageNet-21K**	**789M**@112 + 38.4M@224	1/3	3.4	77.2	58.3
model+data+schedule scaling	**ViT-H/14**	**+ ImageNet-21K**	**789M**@84 + 38.4M@224	2/3	8.1	79.0	60.5
model+data+schedule scaling	**ViT-L/16**	**+ LAION-400M**	**2.56B**@112 + 38.4M@224	1/3	8.8	80.5	62.2
model+data+schedule scaling	**ViT-H/14**	**+ DataComp-1B**	**5.12B**@84 + 38.4M@224	2/3	38.6	83.3	68.2

Table 2. **Scaling behavior of AdvXL**. For each model, we report its training set, the number of training samples it used and their resolution, its PGD number of steps (in pre-training and fine-tuning, respectively), the total training compute (in 1e10 GFLOPS), clean accuracy, and PGD-20-robust accuracy. "+" on the dataset means any additional dataset used during training besides ImageNet-1K. We scale along three aspects: model, data, and scale, and observe consistent improvement in terms of both clean accuracy and robustness.

Dataset	Model	Clean	PGD-20
ImageNet-1K	ViT-B/16	73.0	52.5
	ConvNeXT-B	73.9	54.2
+LAION-400M	ViT-L/16	80.5	62.2
	ConvNeXT-L	77.9	58.5

Table 3. **Architecture comparison** between ViT and ConvNeXT.

Text Encoder	Clean	PGD-20
Base	80.6	62.2
Large	80.5	62.2
Huge	80.6	62.3

Table 4. **CLIP text encoder size**.

ViT-H/14 model on DataComp-1B for 5.12B samples (the last row of Tab. 2), there is not an observed saturation point.

Data scaling. Our AdvXL also exhibits favorable outcomes with web-scale datasets LAION-400M and DataComp-1B. This trend could potentially pave the way for adversarially trained models to rival foundational models like CLIP [47] and Flamingo [1]. Notably, we find that data scaling itself is beneficial, even without a prolonged training schedule. As shown in the second and the fourth rows of Tab. 2, by substituting ImageNet-1K with ImageNet-21K to adversarially train ViT-L/16, we observe an uptick of 1.0% in clean accuracy and a 1.4% increase in robustness, notwithstanding identical training durations. When coupled with our preliminary findings suggesting diminished returns from extended schedules on ImageNet-1K, we conclude that the richness and diversity brought by data scaling stand as pivotal elements in the success of adversarial training at scale.

4.4. Architecture Choice

We have also ablated alternative architectures such as ConvNeXT [37] and Swin-Transformer [35], two leading backbones on RobustBench ImageNet leaderboard [34, 52]. However, our attempts to train a Swin-Transformer with reduced-size inputs posed challenges as the feature size of the last stage may even be smaller than the window size. For example, when employing common configurations like a patch size 4×4 and a window size 7×7, using a 112×112 input would lead to a final stage feature size of 3×3. This mismatch hindered effective training without architectural modifications, and thus, we primarily focus on comparing ViT and ConvNeXT.

To ensure fair evaluation, we maintain consistency with the same two-stage training recipe detailed in Sec. 4.3 during the performance comparison. The results, presented in Tab. 3, demonstrate that ConvNeXT does outperform ViT on a relatively small scale. However, this advantage diminishes as the scale increases, leading us to keep ViT as the default backbone for comparisons against other state-of-the-art models.

Also, we could adopt a larger pre-trained CLIP text encoder in contrastive learning, as it is frozen and introduces little computational overhead. Tab. 4 shows the result of training ViT-L/16 on LAION-400M with various CLIPA text encoders. As can be seen, the performance is robust to a wide range of CLIP text encoder choices. Thus, we simply use the same-scale text encoder to the image encoder (*i.e.* a ViT-L image encoder with a Large text encoder).

4.5. Comparison with SOTA Models

The comparison presented in Tab. 5 evaluates our models against prior works, focusing on l_∞ robustness at ϵ_∞ = 4/255. Following [52], we include l_2 robustness at ϵ_2 = 2 and l_1 robustness at ϵ_1 = 75. Models listed exhibit over 80M parameters and are sorted based on their l_∞ robustness under AutoAttack.

AdvXL emerges as the top performer owing to its unprecedented scale in adversarial training. Our highly efficient two-stage training paradigm facilitates this scala-

Model	Dataset	Samples@Resolution	Pre-trained	Adv. Steps	Params (M)	Compute (1e10)	Source	Clean	l_∞	l_2	l_1
RobArch-L	ImageNet-1K	128M@224		3	104	1.3	[43]	73.5	48.9	39.5	14.7
ViT-B/16		384M@224		2	87	2.7	[49]	76.6	53.5	-	-
ConvNeXT-B		384M@224		3	89	2.4	[34]	76.0	55.8	44.7	21.2
Swin-B		384M@224		3	88	2.4	[34]	76.2	56.2	47.9	23.9
ConvNeXt-B+ConvStem		320M@224	✓	3	89	2.0	[52]	75.2	56.3	49.4	23.6
ConvNeXt-L+ConvStem		128M@224	✓	3	198	1.8	[52]	77.0	57.7	47.0	22.2
ConvNeXt-L+ConvStem		128M@224(320 eval)	✓	3	198	1.8	[52]	78.2	59.4	56.2	33.8
ConvNeXt-L		384M@224		3	198	5.3	[34]	78.0	58.5	-	-
Swin-L		384M@224		3	197	5.3	[34]	78.9	59.6	-	-
ViT-H/14	+ DataComp-1B	5.12B@84 + 38.4M@224 + 6.4M@336		2/3	304	39.6	ours	83.9	69.8	69.8	46.0
ViT-g/14	+ DataComp-1B	5.12B@84 + 38.4M@224 + 6.4M@336		2/3	1013	63.4	ours	83.9	71.0	70.4	46.7

Table 5. **Comparison to SOTA l_∞-robust models on ImageNet**. For each model we report the training set it used, the number and resolution of training samples it used, if it uses pre-trained weights or not, the number of PGD steps in AT (in pre-training and fine-tuning, respectively), the number of parameters of each model, the total training compute (in $1e^{10}$ GFLOPS), its source, its clean accuracy and l_∞, l_2, l_1-robust accuracy with $\epsilon_\infty = 4/255$, $\epsilon_2 = 2$, $\epsilon_1 = 75$(AutoAttack). Note that for the model initialized with pre-trained weight, the pre-training compute is not included. For unavailable metrics of those publicly unavailable models, we use "-" to fill in the blank. "+" on the dataset means any additional dataset used during training besides ImageNet-1K. Our AdvXL successfully secures new state-of-the-art records on all three robustness metrics thanks to its unprecedented model and data scale.

bility without incurring excessive computational expenses. For instance, our largest ViT-g/14 model trained on the DataComp-1B dataset achieves outstanding results with a computing budget of merely about $12\times$ that of the previous best results from [34]. Despite this relatively modest computational investment, our model outperforms them by an impressive 11.4% in terms of l_∞-robust accuracy under AutoAttack. We would like to stress that training with full resolution and strong attacks on 5.12B samples, without our efficiency design, would incur $\sim 20\times$ the computational cost of our approach (equating to $\sim 250\times$ the compute of the previous best results), rendering such an endeavor computationally infeasible.

Even more noteworthy is the exceptional generalizability showcased by our AdvXL ViT-g/14 models trained on the web-scale DataComp-1B dataset, securing l_2-robust accuracy of 70.4% and l_1-robust accuracy of 46.7%. These figures represent an absolute improvement of about 13% over the best previous results. This observation indicates that scaling model, data, and schedule collectively not only significantly enhances robustness against known attacks but also fortifies the model against unseen attacks during training. Our findings on scaling adversarial training illuminate the path towards the evolution of next-generation robust visual models, potentially propelling the field of adversarial training into the era of foundation models.

5. Discussion and Conclusion

Adversarial training has traditionally been confined to small networks and datasets, predominantly ResNet-50 and CIFAR-10. Until recently, there have been few attempts to train adversarially robust models on the medium-size ImageNet-1K dataset. In this work, we break new ground by scaling adversarial training to web-scale datasets containing over 1B samples. Our AdvXL approach comprises two core components: 1) a coarse-to-fine, weak-to-strong, two-stage training paradigm to mitigate the computational cost of scaling up; 2) the utilization of a pre-trained CLIP text encoder enabling training on web-scale datasets. Through scaling along model, data, and schedule dimensions, we successfully establish a new state-of-the-art record of l_∞-robust accuracy under AutoAttack, surpassing the previous best by a margin of $\sim 10\%$. Additionally, training on those gigantic datasets demonstrates increased generalizability against unseen attacks during training, aligning with observations from various foundation models [1, 6, 45–47]. We envision our work as a stepping stone for adversarial training to enter the era of foundation models, inspiring further large-scale adversarial training endeavors.

Broad impact. Our method delivers over $5\times$ speedup, significantly reducing wall-clock time for training models with hundreds of millions or even billions of parameters on billion-scale datasets (*e.g.*, on the order of thousands of TPU/GPU-days). AdvXL not only facilitates rapid prototyping and accelerated research cycles but also contributes to substantial energy and carbon emissions savings, a critical consideration in large-scale model training.

Acknowledge

This work is partially supported by a gift from Open Philanthropy. We thank Center for AI Safety, TPU Research Cloud (TRC) program, and Google Cloud Research Credits program for supporting our computing needs.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *NeurIPS*, 2022. 7, 8

[2] Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. In *ECCV*, 2020. 5

[3] Yutong Bai, Jieru Mei, Alan L Yuille, and Cihang Xie. Are transformers more robust than cnns? *NeurIPS*, 2021. 2

[4] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEit: BERT pre-training of image transformers. In *ICLR*, 2022. 3

[5] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *ECML PKDD*, 2013. 3

[6] 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. *NeurIPS*, 2020. 1, 8

[7] Tianlong Chen, Sijia Liu, Shiyu Chang, Yu Cheng, Lisa Amini, and Zhangyang Wang. Adversarial robustness: From self-supervised pre-training to fine-tuning. In *CVPR*, 2020. 2

[8] Francesco Croce and Matthias Hein. Minimally distorted adversarial examples with a fast adaptive boundary attack. In *ICML*, 2020. 5

[9] Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. In *NeurIPS*, 2021. 2, 5

[10] Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. In *NeurIPS*, 2020. 5

[11] Edoardo Debenedetti, Vikash Sehwag, and Prateek Mittal. A light recipe to train robust vision transformers. In *SaTML*, 2023. 2

[12] 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 *ICML*, 2023. 1, 2

[13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 4, 6

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019. 1

[15] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 5

[16] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *CVPR*, 2023. 1, 2

[17] Yonggan Fu, Shunyao Zhang, Shang Wu, Cheng Wan, and Yingyan Lin. Patch-fool: Are vision transformers always robust against adversarial perturbations? In *ICLR*, 2022. 2

[18] Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. In *NeurIPS*, 2022. 4, 6

[19] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *ICLR*, 2015. 1, 2, 3

[20] Sven Gowal, Chongli Qin, Jonathan Uesato, Timothy Mann, and Pushmeet Kohli. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593*, 2020. 2

[21] Jindong Gu, Volker Tresp, and Yao Qin. Are vision transformers robust to patch perturbations? In *ECCV*, 2022. 2

[22] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. In *ICLR*, 2022. 4

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1

[24] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 3, 5

[25] Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model robustness and uncertainty. In *ICML*, 2019. 2

[26] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack William Rae, and Laurent Sifre. An empirical analysis of compute-optimal large language model training. In *NeurIPS*, 2022. 6

[27] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *ECCV*, 2016. 5

[28] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009. 1

[29] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and Rene Ranftl. Language-driven semantic segmentation. In *ICLR*, 2022. 4

[30] Xianhang Li, Zeyu Wang, and Cihang Xie. Clipa-v2: Scaling clip training with 81.1 *arXiv preprint arXiv:2306.15658*, 2023. 3, 4

[31] Xianhang Li, Zeyu Wang, and Cihang Xie. An inverse scaling law for clip training. In *NeurIPS*, 2023. 3, 4, 5

[32] Yanghao Li, Haoqi Fan, Ronghang Hu, Christoph Feichtenhofer, and Kaiming He. Scaling language-image pre-training via masking. In *CVPR*, 2023. 3, 4, 5, 6

[33] Zichao Li, Li Liu, Zeyu Wang, Yuyin Zhou, and Cihang Xie. Bag of tricks for fgsm adversarial training. *arXiv preprint arXiv:2209.02684*, 2022. 2

[34] Chang Liu, Yinpeng Dong, Wenzhao Xiang, Xiao Yang, Hang Su, Jun Zhu, Yuefeng Chen, Yuan He, Hui Xue, and Shibao Zheng. A comprehensive study on robustness of image classification models: Benchmarking and rethinking. *arXiv preprint arXiv:2302.14301*, 2023. 2, 7, 8

[35] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 2, 7

[36] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *CVPR*, 2022. 2

[37] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *CVPR*, 2022. 2, 7

[38] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 3, 5

[39] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018. 1

[40] Yichuan Mo, Dongxian Wu, Yifei Wang, Yiwen Guo, and Yisen Wang. When adversarial training meets vision transformers: Recipes from training to architecture. *NeurIPS*, 2022. 2

[41] OpenAI. Gpt-4 technical report. *ArXiv*, abs/2303.08774, 2023. 1, 2

[42] Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu. Bag of tricks for adversarial training. In *ICLR*, 2021. 2

[43] ShengYun Peng, Weilin Xu, Cory Cornelius, Kevin Li, Rahul Duggal, Duen Horng Chau, and Jason Martin. Robarch: Designing robust architectures against adversarial attacks. *arXiv preprint arXiv:2301.03110*, 2023. 8

[44] Omid Poursaeed, Isay Katsman, Bicheng Gao, and Serge Belongie. Generative adversarial perturbations. In *CVPR*, 2018. 3

[45] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. *OpenAI blog*, 2018. 8

[46] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 2019.

[47] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 3, 4, 5, 7, 8

[48] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. In *CVPR*, 2022. 4

[49] Sylvestre-Alvise Rebuffi, Francesco Croce, and Sven Gowal. Revisiting adapters with adversarial training. In *ICLR*, 2023. 2, 8

[50] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. 4, 6

[51] Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! *NeurIPS*, 32, 2019. 2, 3

[52] Naman D Singh, Francesco Croce, and Matthias Hein. Revisiting adversarial training for imagenet: Architectures, training and generalization across threat models. In *NeurIPS*, 2023. 2, 7, 8

[53] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *ICLR*, 2014. 3

[54] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *ECCV*, 2020. 4

[55] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 1

[56] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 1

[57] Zekai Wang, Tianyu Pang, Chao Du, Min Lin, Weiwei Liu, and Shuicheng Yan. Better diffusion models further improve adversarial training. In *ICML*, 2023. 2

[58] Eric Wong, Leslie Rice, and J. Zico Kolter. Fast is better than free: Revisiting adversarial training. In *ICLR*, 2020. 2

[59] Chaowei Xiao, Bo Li, Jun-Yan Zhu, Warren He, Mingyan Liu, and Dawn Song. Generating adversarial examples with adversarial networks. *IJCAI*, 2018. 3

[60] Cihang Xie and Alan Yuille. Intriguing properties of adversarial training at scale. In *ICLR*, 2020. 2

[61] Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L. Yuille, and Kaiming He. Feature denoising for improving adversarial robustness. In *CVPR*, 2019.

[62] Cihang Xie, Mingxing Tan, Boqing Gong, Alan Yuille, and Quoc V Le. Smooth adversarial training. *arXiv preprint arXiv:2006.14536*, 2020. 2

[63] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019. 5

[64] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In *CVPR*, 2022. 1, 2

[65] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018. 5

[66] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. Regionclip: Region-based language-image pretraining. In *CVPR*, 2022. 4

[67] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *ECCV*, 2022. 4