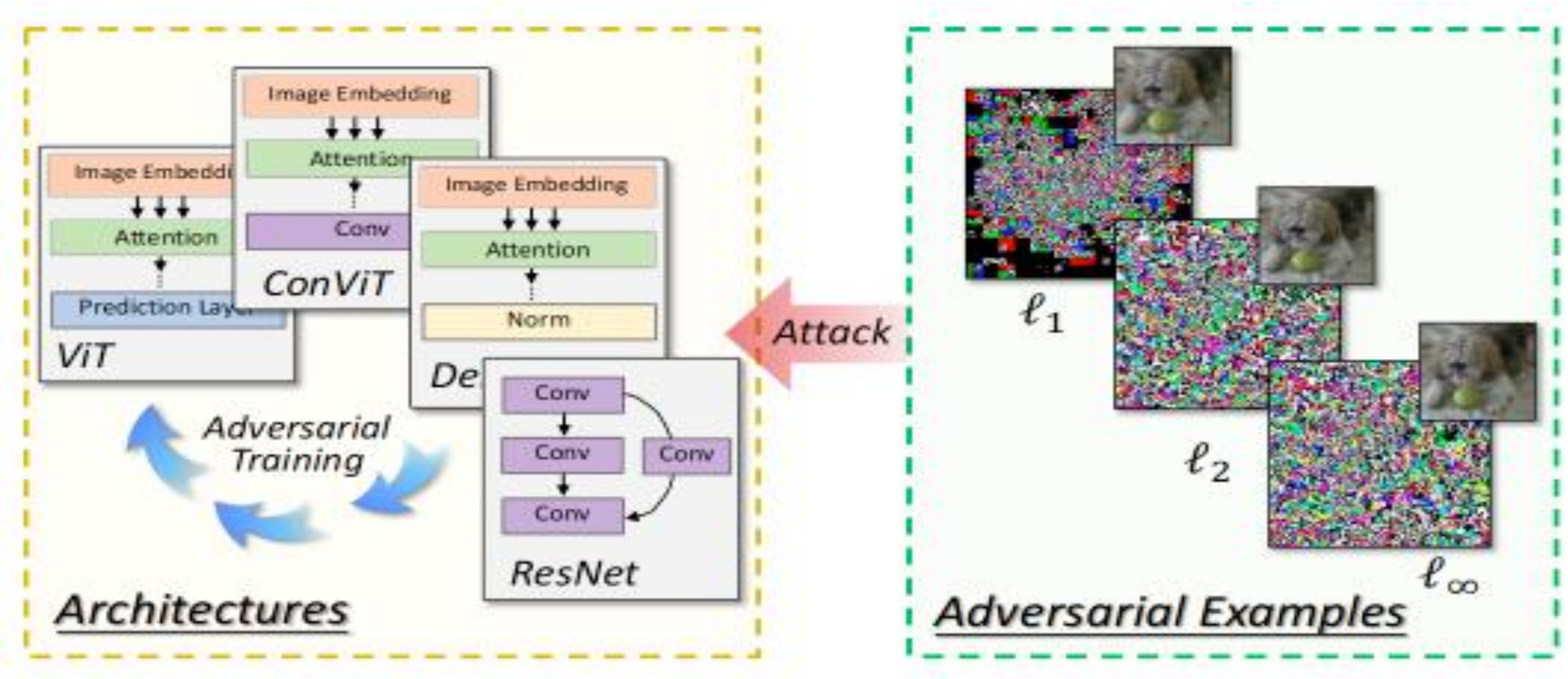
Exploring the Relationship between Architectural Design and Adversarially Robust Generalization

Aishan Liu[1], Shiyu Tang[1], Siyuan Liang[2], Ruihao Gong[1,6], Boxi Wu[3], Xianglong Liu[1,4,5]①, Dacheng Tao[7]
[1]Beihang University, [2]Chinese Academy of Sciences, [3]Zhejiang University, [4]Zhongguancun Laboratory, [5]Hefei Comprehensive National Science Center, [6]SenseTime, [7]JD Explore Academy

CONTENTS

- **Connection between Network Structure with Generalization ?**
- **Adversarially Robust Generalization ?**



CONTENTS

Contribution

- We, for the first time, systematically studied 20 adversarially-trained architectures against multiple attacks and revealed the close relationship between architectural design and robust generalization.
- We theoretically revealed that higher weight sparsity contributes to the better adversarially robust generalization of Transformers, which can often be achieved by attention blocks.
- We provide more detailed analyses of the generalizability from several viewpoints and discuss potential pathways that may improve architecture robustness.

CONTENTS

Empirical Evaluation

- **Datasets**： CIFAR-10 and ImageNette
- **Architectures**： CNN with convolutions, ViT with attentions, hybrids with both attention/convolutions, and newly designed atten_x0002_tions), aiming to find the influential parts.
- **Training settings**： Standard training (vanilla training) and PGD-l_∞ adversarial training
- **Evaluation strategy**： $W(f_{\boldsymbol{\theta}}, \mathbb{A}) = \frac{1}{n} \sum_{i=1}^{n} \left\{ \min_{\mathcal{A} \in \mathbb{A}} \mathbf{1}\left[f_{\boldsymbol{\theta}}\left(\mathcal{A}\left(\boldsymbol{x}_i \right) \right) = \boldsymbol{y}_i \right] \right\}$

Architecture	Params (M)	Vanilla Acc	PGD-ℓ_∞ Adversarial Training					
			Clean Acc	PGD-ℓ_∞	AA-ℓ_∞	PGD-ℓ_2	PGD-ℓ_1	Worst-case Acc
PVTv2	12.40	88.34	75.99	46.48	38.18	35.77	46.14	33.54
CoAtNet	16.99	90.73	77.73	48.27	39.85	33.80	42.30	32.17
ViT	9.78	86.73	78.76	46.02	38.00	30.86	39.27	29.24
CPVT	9.49	90.34	78.57	45.02	36.73	30.15	39.22	28.47
ViTAE	23.18	88.24	75.42	40.53	33.22	29.67	40.02	28.13
MLP-Mixer	0.68	83.43	62.86	38.93	31.81	29.27	36.50	27.42
PoolFormer	11.39	89.26	73.66	46.33	38.93	28.84	34.32	27.36
CCT	3.76	92.27	81.23	49.21	40.97	28.29	34.59	26.82
VGG	14.72	94.01	84.30	50.87	41.66	26.78	31.48	25.32
Swin Transformer	27.42	91.58	80.44	48.61	41.31	26.58	30.47	25.04
LeViT	6.67	89.01	77.10	47.16	39.87	26.28	29.58	25.04
MobileViT	5.00	91.47	77.52	49.51	41.50	26.96	29.35	24.41
BoTNet	18.82	94.16	80.76	51.29	42.95	25.84	27.38	23.15
WideResNet	55.85	96.47	89.54	55.17	44.13	22.55	23.68	20.88
DenseNet	1.12	94.42	83.23	53.06	44.02	22.55	21.87	19.48
PreActResNet	23.50	95.86	87.96	54.85	45.81	18.60	16.46	15.11
CeiT	5.56	85.24	71.55	36.20	28.02	15.31	16.77	14.35
ResNet	23.52	95.60	87.92	54.18	45.40	17.52	15.90	14.32
ResNeXt	9.12	95.64	87.12	51.51	42.66	15.07	13.64	12.18
CvT	19.54	87.81	73.76	41.36	33.67	12.75	9.25	8.76

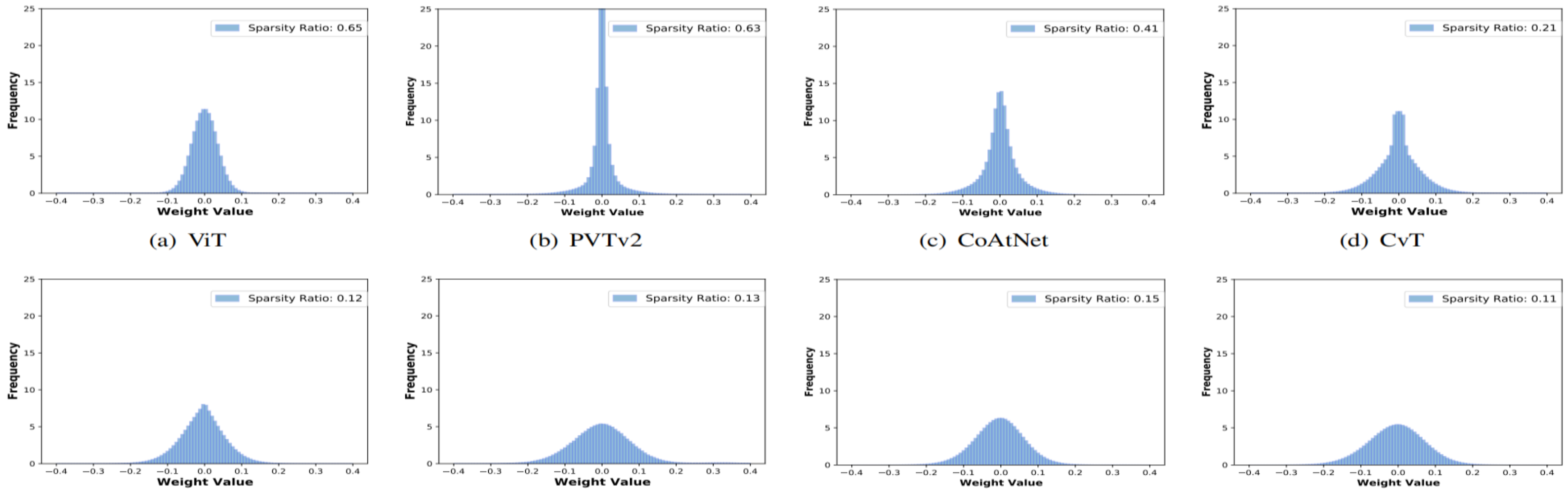
Overall understanding: weight sparsity

- Rademacher complexity

$$R_{\mathcal{S}}(\mathcal{F}) = \frac{1}{n}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}}\sum_{i=1}^{n}\sigma_i f(\boldsymbol{x}_i)\right]$$

- Lemma 1

$$\frac{c}{2}\left(R_{\mathcal{S}}(\mathcal{F}) + \epsilon\Theta\frac{d^{1-\frac{1}{p}}}{\sqrt{n}}\right) \leqslant R_{\mathcal{S}}(\hat{\mathcal{F}}) \leqslant R_{\mathcal{S}}(\mathcal{F}) + \epsilon\Theta\frac{d^{1-\frac{1}{p}}}{\sqrt{n}}$$



(a) ViT (b) PVTv2 (c) CoAtNet (d) CvT

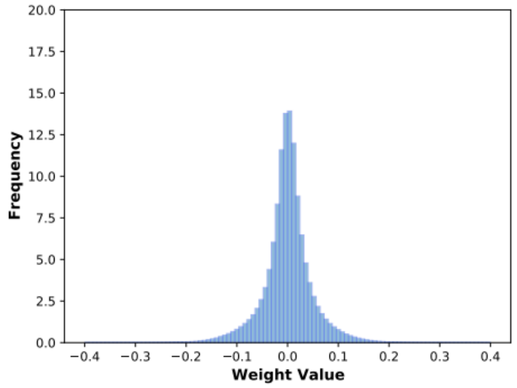
(e) VGG (f) DenseNet (g) ResNet (h) WideResNet

Attention contributes to sparseness

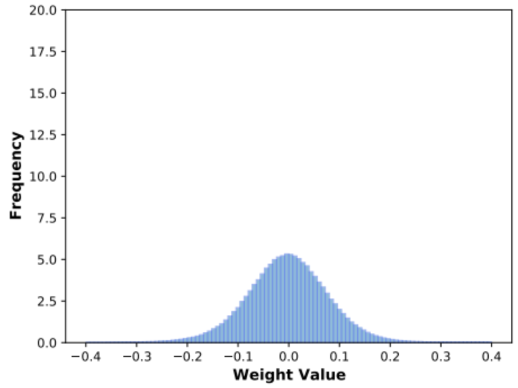
Theorem 1. *Suppose the Transformer network function* $\mathcal{F} = \{f_{\boldsymbol{w}}(x) \ : \boldsymbol{W} = (\boldsymbol{A}_1, \boldsymbol{W}_1), \| \boldsymbol{A}_1 \|_p \leq s_1, \| \boldsymbol{W}_1 \|_p \leq s_2, \| \boldsymbol{A}_1 \|_1 \leq b_1, \| \boldsymbol{W}_{1,1} \|_1 \leq b_2\}$*.* $\forall \gamma > 0$*, with probability at least* $1 - \delta$*, we have* $\forall f_{\boldsymbol{w}} \in \mathcal{F}$,

$$\begin{aligned} &\mathbb{P}_{(\boldsymbol{x},y)\sim\mathcal{S}}\{\exists \boldsymbol{\delta} \in \mathbb{B}(\epsilon), \quad \text{s.t.} \quad \boldsymbol{y} \neq \arg\max[f_{\boldsymbol{w}}(\boldsymbol{x} + \boldsymbol{\delta})]_{\boldsymbol{y}'}\} \\ &\leq \frac{1}{n}\sum_{i=1}^{n} E_i + \frac{1}{\gamma}\left(\frac{4}{n^{3/2}} + \frac{60\log(n)\log(2d_{\max})}{n} s_1 s_2 C\right) \\ &+ \frac{2\epsilon b_1 b_2}{\gamma\sqrt{n}} + 3\sqrt{\frac{\log(2/p)}{2n}}, \end{aligned} \tag{6}$$

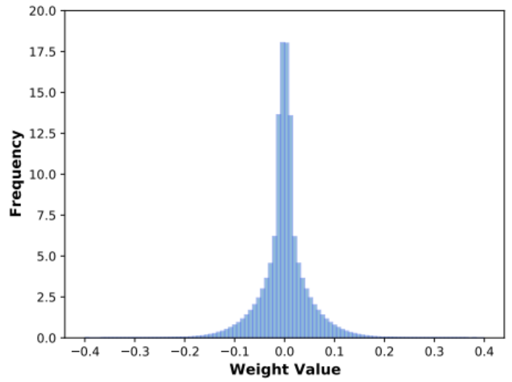
where $\boldsymbol{w}_{1,k}$ denotes the k-th column of $\boldsymbol{W}_1$, $C = ((\frac{b_1}{s_1})_{2/3} + (\frac{b_2}{s_2})_{2/3})_{3/2} \| \boldsymbol{X} \|_F$, $E_i = \mathbb{1}([f_{\boldsymbol{w}}(\boldsymbol{x}_i)]_{\boldsymbol{y}_i} + \frac{\epsilon}{2}\max_{k\in[K], z=\pm 1 \boldsymbol{P}\succeq 0, diag\boldsymbol{P}\leq 1}\langle zQ(\boldsymbol{w}_{1,k}, \boldsymbol{A}_1), \boldsymbol{P}\rangle)$, and $Q(\boldsymbol{w}_{1,k}, \boldsymbol{A}_1) = \begin{bmatrix} 0 & 0 & \boldsymbol{1}^\top \boldsymbol{A}_1^\top \boldsymbol{b} \\ 0 & 0 & \boldsymbol{A}_1^\top \boldsymbol{b} \\ \boldsymbol{b}^\top \boldsymbol{A}_1 \boldsymbol{1} & \boldsymbol{b}^\top \boldsymbol{A}_1 & 0 \end{bmatrix}$. At this time, there is $\boldsymbol{b} = \mathrm{diag}(\boldsymbol{w}_{1,k})$.



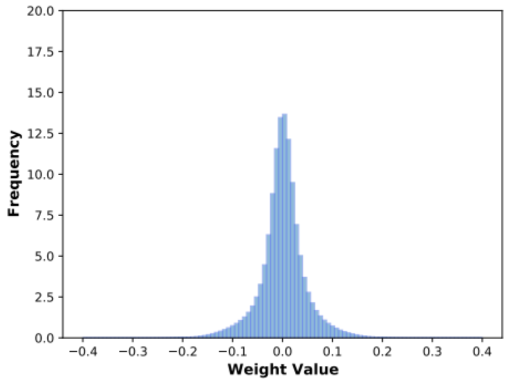
(a) All Weights



(b) Conv Weights



(c) QKV Weights

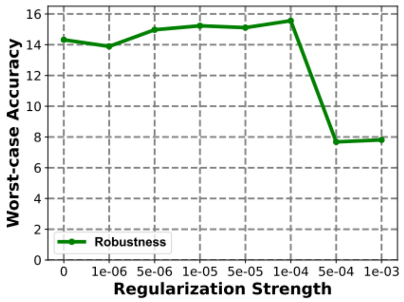


(d) Feedforward Weights

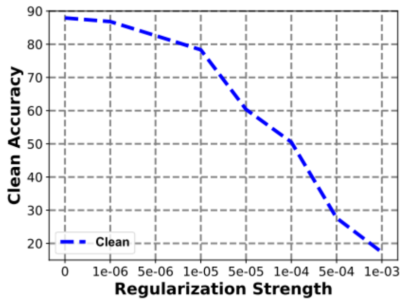
CONTENTS

Potential Pathways

- Imposing l_1 regularization for sparsity.



(a) Worst-case



(b) Clean

- Hybrid architecture with increased sparsity.

			PGD-ℓ_∞ Adversarial Training					
Architecture	**Params (M)**	**Vanilla**	**Clean**	**PGD-ℓ_∞**	**AA-ℓ_∞**	**PGD-ℓ_2**	**PGD-ℓ_1**	**Worst-case**
CoAtNet-CTTT	17.36	90.83	74.25	45.70	37.27	31.95	40.15	30.30
CoAtNet-CTTC	17.02	91.13	77.84	46.61	37.89	31.61	39.88	29.96
CoAtNet-CTCC	16.38	90.69	78.71	42.36	34.56	27.84	37.19	26.69
CoAtNet-CCCC	16.02	91.41	79.14	43.71	35.59	29.03	38.68	27.64

- Patch size as receptive fields.

Architecture	Patch Size	Vanilla	PGD-ℓ_∞ Adversarial Training					
			Clean	PGD-ℓ_∞	AA-ℓ_∞	PGD-ℓ_2	PGD-ℓ_1	Worst-case
PVTv2	$p = 4$	88.34	75.99	46.48	38.18	35.77	46.14	33.54
	$p = 2$	93.03	83.80	52.34	44.04	32.49	39.63	31.16
	$p = 1$	94.60	87.50	54.59	46.58	23.47	24.76	21.10
ViT	$p = 8$	82.30	72.39	42.77	35.04	32.74	42.61	30.72
	$p = 4$	86.73	78.76	46.02	38.00	30.86	39.27	29.24
	$p = 2$	85.99	77.37	45.45	37.95	25.36	30.15	23.78

- Considering generalization on common corruptions.

Architecture	CIFAR-10 Dataset			
	Vanilla Training		PGD-ℓ_∞ Training	
	Vanilla Acc	CIFAR-C Acc	Clean Acc	CIFAR-C Acc
WideResNet	96.47	83.91	89.54	81.48
ResNet	95.60	81.20	87.92	79.24
PreActResNet	95.86	82.18	87.96	78.99
ResNeXt	95.64	80.43	87.12	77.76
VGG	94.01	81.22	84.30	75.85
DenseNet	94.42	79.73	83.23	74.60
BoTNet	94.16	81.04	80.76	72.72
CCT	92.27	78.99	81.23	72.56
ViT	86.73	77.06	78.76	71.87
Swin Transformer	91.58	77.61	80.44	71.36
CPVT	90.34	79.66	78.57	70.74
LeViT	89.01	78.31	77.10	70.48
CoAtNet	90.73	79.91	77.73	70.27
MobileViT	91.47	80.48	77.52	70.15
PVTv2	88.34	79.84	75.99	69.12
ViTAE	88.24	75.86	75.42	67.58
PoolFormer	89.26	77.57	73.66	66.45
CVT	87.81	75.10	73.76	66.28
CeiT	85.24	73.99	71.55	65.07
MLP-Mixer	83.43	70.70	62.86	57.09

Thank You!

Email

liuaishan@buaa.edu.cn