REGroup: Rank-aggregating Ensemble of Generative Classifiers for Robust Predictions

Lokender Tiwari[1,2] , **Anish Madan**[2] , **Saket Anand**[2] , **Subhashis Banerjee**[3,4]

[1]TCS Research, [2]IIIT-Delhi, [3]IIT Delhi, [4]Dept. of Computer Science, Ashoka University

Project Page: https://lokender.github.io/REGroup.html

Problem Definition and Contribution

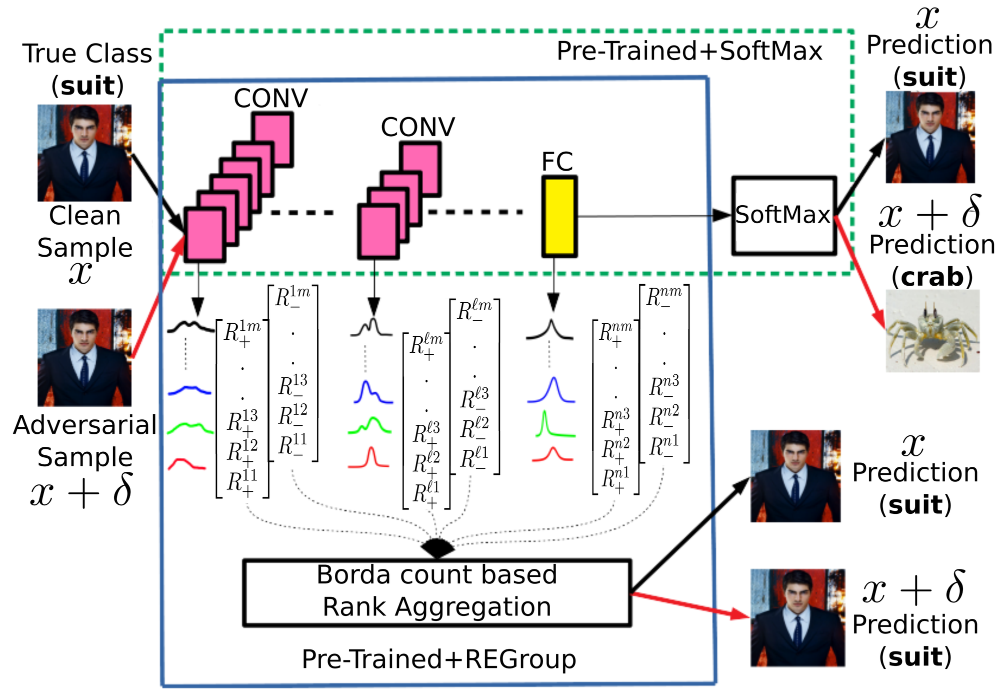
Goal: Defending against adversarial attacks in deep neural networks without expensive adversarial training or fine-tuning

Our Approach: Learn class conditional ***generative*** classifiers by statistically characterizing the ***pre-activation*** neural responses of intermediate layers to clean training samples

- Make ranked predictions at intermediate layers using generative classifiers
- Aggregate the ranked predictions from the intermediate-layers using Borda-count[2] to make final predictions

Key Advantages:

- Make a pre-trained classifier robust to adversarial attacks
- *Agnostic to* : adversarial attacks, classifier architectures
- *Scalable* : ImageNet, CIFAR10



REGroup Methodology

Layerwise Neural Response Distributions: We model layerwise PMFs of neuronal responses using the pre-activation feature maps for a subset S of the training set.

- We denote the PMFs by $\mathbb{P}_j^{\ell i}$ and $\mathbb{N}_j^{\ell i}$ corresponding to positive and negative responses. Here ℓ, i and j denote the ℓ^{th} layer, i^{th} feature map and the j^{th} input sample respectively.

Layerwise Generative Classifiers: We model the layerwise generative classifiers for class y as a class-conditional mixture of distributions, with each mixture component as the PMFs $\mathbb{P}_j^{\ell}$ and $\mathbb{N}_j^{\ell}$ for a given training sample $x_j \in \mathcal{S}_y$.

$$\mathbb{C}_y^{+\ell} = \sum_{j:x_j \in \mathcal{S}_y} \lambda_j \mathbb{P}_j^{\ell}, \qquad \mathbb{C}_y^{-\ell} = \sum_{j:x_j \in \mathcal{S}_y} \lambda_j \mathbb{N}_j^{\ell} \tag{1}$$

At inference time, we compute the PMFs $\mathbb{P}_j^{\ell}$ and $\mathbb{N}_j^{\ell}$ for a test sample x_j. Then, we compute KL-Divergence between the classifier model $\mathbb{C}^{+\ell}$ and the test sample $\mathbb{P}_j^{\ell}$ (and similarly for $\mathbb{N}_j^{\ell}$) as a classification score:

$$P_{KL}(\ell, y) = \sum_i \mathbb{C}_y^{+\ell i} \log \left(\frac{\mathbb{C}_y^{+\ell i}}{\mathbb{P}^{\ell i}} \right), \forall y \in \{1, \dots, M\} \tag{2}$$

Rank Ordering and Aggregation: We rank-order the classes, which we simply achieve by sorting the KL-Divergences (Eqn. (2)) in ascending order. $R_+^{\ell y}$ is the rank of y^{th} class in the ℓ^{th} layer preference list R_+^{ℓ}.

$$R_+^{\ell} = [R_+^{\ell 1}, R_+^{\ell 2}, ..., R_+^{\ell y}, ..., R_+^{\ell M}], \quad R_-^{\ell} = [R_-^{\ell 1}, R_-^{\ell 2}, ..., R_-^{\ell y}, ..., R_-^{\ell M}] \tag{3}$$

- The individual layer's class ranking preferences are aggregated using Borda count-based scoring. The individual Borda count of both voters are denoted by $B_+^{\ell y}$ and $B_-^{\ell y}$ and M is the number of classes.

$$B_+^{\ell y} = (M - R_+^{\ell y}), \qquad B_-^{\ell y} = (M - R_-^{\ell y}); \tag{4}$$

- We aggregate the Borda counts of highest k layers of the network. Let $B^{:ky}$ denote the aggregated Borda count of y^{th} class from the last k layers. Our final prediction is denoted by $\widehat{y}$.

$$B^{:ky} = \sum_{\ell=n-k+1}^{n} B^{\ell y} = \sum_{\ell=n-k+1}^{n} B_+^{\ell y} + B_-^{\ell y}, \ \ \forall y \in \{1..M\}; \quad \widehat{y} = argmax_y \ B^{:ky}$$

References

[1] Edward Raff et al. "Barrage of Random Transforms for Adversarially Robust Defense". In: *CVPR*. 2019.

[2] Jörg Rothe. "Borda Count in Collective Decision Making: A Summary of Recent Results". In: *AAAI*. 2019.

[3] Cihang Xie et al. "Feature denoising for improving adversarial robustness". In: *CVPR*. 2019.

Experiments & Results

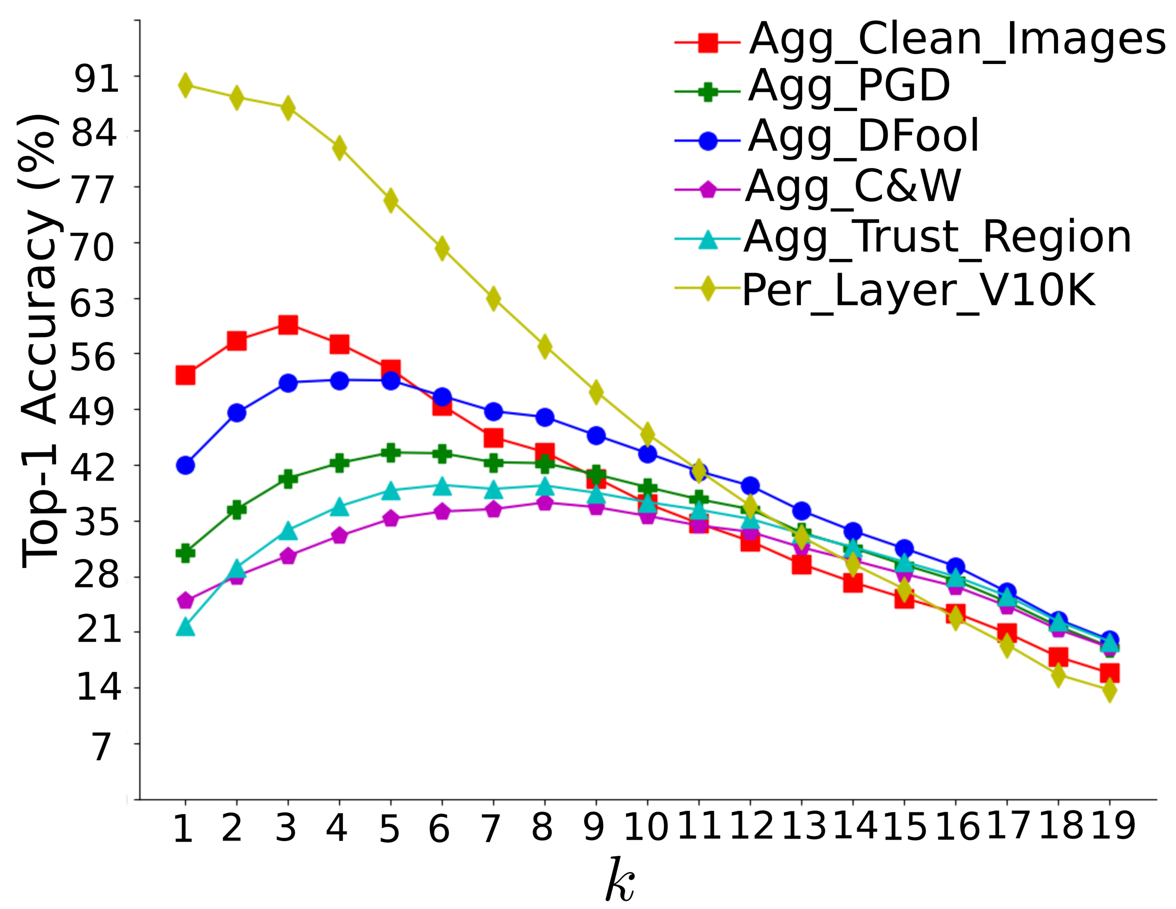
Network Architectures & Dataset:

- **Network architectures.** We consider ResNet-50 and VGG-19 architectures, pre-trained on ImageNet dataset.
- **Dataset.** We present our evaluations, comparisons and analysis on the ImageNet dataset. We use the subsets of full ImageNet validation set as described in Tab. 1.

Dataset	Description
V50K	Full ImageNet validation set with 50000 images.
V10K	A subset of 10000 correctly classified images from V50K set. 10 Per class.
V2K	A subset of 2000 correctly classified images from V50K set. 2 Per class.
V10C	A subset of correctly classified images of 10 sufficiently different classes.

Table 1: Dataset used for evaluation and analysis

Accuracy vs no. of layers



Comparison with adversarially trained and fine-tuned classification models

(Dataset used: V50K).	Clean Images		Attacked Images	
Model	Top-1	Top-5	Top-1	Top-5
ResNet-50	76	93	0.0	0.0
Inception v3	78	94	0.7	4.4
ResNet-152	79	94	-	-
Inception v3 w/Adv. Train	78	94	1.5	5.5
ResNet-152 w/Adv. Train [3]	63	-	45	-
ResNet-152 w/Adv. Train [3]w/ denoise	66	-	49	-
ResNet-50-BaRT [1], $\hat{k}=5$	65	85	16	51
ResNet-50-BaRT [1], $\hat{k}=10$	65	85	36	57
ResNet-50-REGroup	66	86	22	65

Table 2: The results are divided into three blocks, the top block include original networks, middle block include defense approaches based on adversarial re-training/fine-tuning of original networks, bottom block is our defense *without re-training/fine-tuning*.

Performance on Gradient-Free Attacks:

			ResNet-50		VGG-19	
			REGroup		REGroup	
	Data	ϵ	#S	T1(%)	#S	T1(%)
SPSA	V10K	$4\ (L_\infty)$	4911	71	5789	58
Boundary	V10K	$2\ (L_2)$	10000	50	10000	50
Spatial	V10K	$2\ (L_2)$	2624	36	2634	30

Table 3: Top-1 (%) classification accuracy for REGroup. Note that top-1 accuracy for all cases of softmax are 0.

Performance on Gradient Based Attacks:

				ResNet-50			VGG-19		
		UN /			SMax	REGroup		SMax	REGroup
	Data	TA / HC	ϵ	#S	T1(%)	T1(%)	#S	T1(%)	T1(%)
Clean	V10K	–	–	10000	100	88	10000	100	76
Clean	V2K	–	–	2000	100	86	2000	100	72
Clean	V10C	–	–	417	100	84	392	100	79
PGD	V10K	UN	$4\ (L_\infty)$	9997	0	48	9887	0	46
C&W	V10K	UN	$4\ (L_2)$	10000	0	40	10000	0	38
cAdv	V10C	UN	–	417	0	37	392	0	18
PGD	V2K	TA	(L_∞)	2000	0	47	2000	0	31
PGD	V2K	UN+HC	(L_∞)	2000	0	21	2000	0	19

Table 4: **Performance on Gradient-Based Attacks.** Comparison of Top-1 classification accuracy between SoftMax (SMax) and REGroup based final classification. **Notation**:UN -> Untargeted Attack, TA: Targeted Attack(selects target class randomly), HC: High Confidence (> 90% confidence, and ϵ is unbounded).