Transferring Adversarial Robustness Through Robust Representation Matching

Pratik Vaishnavi, Kevin Eykholt, Amir Rahmati

August 2022

Intro

- ML models can be fooled by carefully crafted adversarial examples
- Need ways to make models robust to such adversarial attacks
- Existing defensive measures are often poorly suited for real world use
- This work proposes a mechanism for transferring the adversarial robustness between models

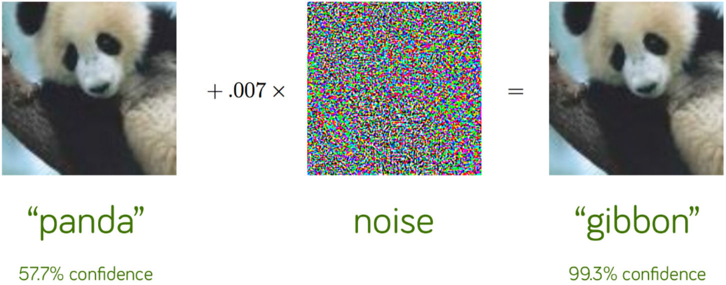

Figure 1: ML algorithms and especially DNNs are often brittle.

Standard Training

- Empirical Risk Minimization (ERM) updates the parameters, θ, of a ANN, F_θ, to minimize the learning model's loss, L

$$\min_\theta L(F_\theta(x), y)$$

Adversarial Attacks

- Adversarial Evasion Attacks (AEA) attempt to imperceptibly perturb inputs to cause misclassification
- Adversaries objective is to add a small perturbation, $\delta < \epsilon$, that maximizes the model's loss

$$\max_{\delta} L(F_{\theta}(x + \delta), y)$$

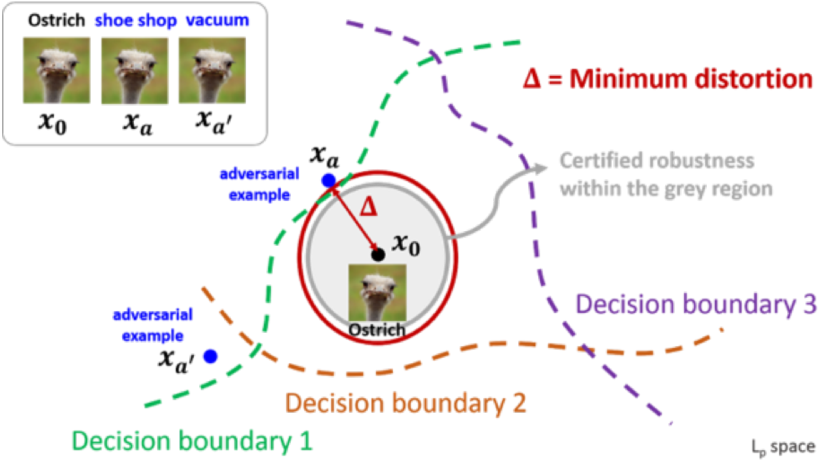

Figure 2: AEAs minimally perturb inputs to attain incorrect classification.

Adversarial Defense

- Adversarial Training (AT) is the best method of defense
- Attempts to find parameters that minimize the adversary's expected attempts to increase loss

$$\min_{\theta} \max_{\delta} L(F_{\theta}(x + \delta), y)$$

- Essentially, augments the training data with adversarial inputs
- Requires several forward-backward passes at each iteration vs a single pass

Explaining Robustness

- Adversarial examples are effective because of a model's tendency to learn non-robust features
- Robust models must learn to focus on robust features that are strongly correlated with the input label
- Knowledge of robust features could be transferred between models

Transferring Adversarial Robustness

- Model robustification should not:
    1. reduce performance on non-adversarial examples
    2. be cost prohibitive
- Transferring robustness can eliminate the need to perform AT during retraining and make robustification cost efficient

Robust Representation Matching

- Robust Representation Matching (RRM) uses a student-teacher framework to transfer the knowledge of feature importance between models
- Trains a teacher model with AT
- Trains a student model with combined objective:
  1. Minimize the cross-entropy loss, L_C
  2. Minimize the robust representation loss, L_R

Robust Representation Matching (cont)

- Formally, the training objective for determining the parameters, θ, of the student NN S_θ is

$$\min_\theta \left[\lambda \cdot L_C(S_\theta(x), y) + L_R(x) \right]$$

- where the robust representation loss is the distance, e.g., cosine similarity, between output of the penultimate layers of the student and teacher models

$$L_R(x) = d(g_S(x), g_T(x))$$

- and λ weighs the contribution of the two different objectives

Why Match the Penultimate Layer?

- Including the robust representation loss term L_R forces the student to match the teacher's penultimate layer
- Matching the penultimate layer can transfer more knowledge than matching the output layer and is architecture-agnostic

Adversarial Training Speedup

- When compared against other AT methods:
  - RRM achieves comparable performance to SAT/Fast AT in significantly less training time
  - RRM achieves greater performance to Free AT in almost the same training time

Method	Training Time	Natural Accuracy	Adversarial Accuracy
SAT	1808	86%	48%
Fast AT	193	84%	50%
Free AT	29	71%	42%
RRM	30	76%	49%

Adversarial Robustness Transfer

- When compared against other transfer methods, RRM vastly outperforms its competitors

Method	Natural Accuracy	Adversarial Accuracy
RDT	80%	1%
KD	83%	3%
RRM	81%	46%

Tuning λ

- Recall RRM's optimization objective:

$$\min_{\theta}\left[\lambda \cdot L_C(S_\theta(x), y) + L_R(x)\right]$$

- L_C encourages the model to learn natural accuracy
- L_R encourages the model to learn robust representations
- λ balances the two training objectives

Tuning λ (cont)

- Increasing λ increases the importance of L_C and increases natural accuracy
- Decreasing λ increases the importance of L_R and increases adversarial accuracy (to an extent)

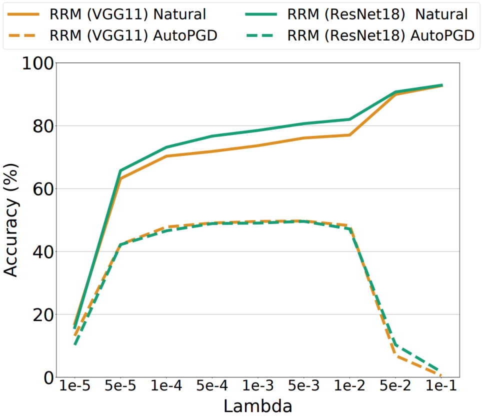

Figure 3

Limit Testing

- Hypothesize that training time per epoch roughly approximates a model's expressive power
- Found that simpler students struggle to learn from complex teachers because they are not complex enough to learn the robust features of the teacher

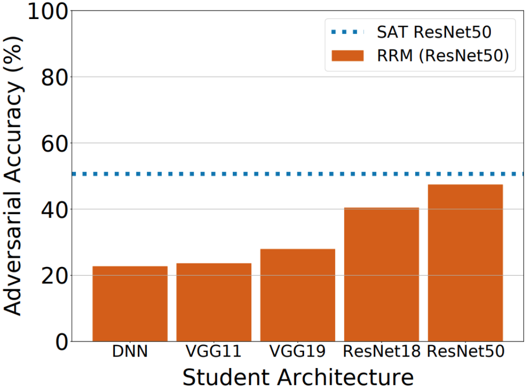

Figure 4

Limitations and Future Work

- RRM still depends on a teacher model and the difficulties that go along with using AT to attain one
- This work only studies RRM with respect to DNNs and image classification

Conclusions

- Introduced Robust Representation Matching (RRM) technique to transfer robustness between DNN models
- Demonstrated that RRM outperforms other adversarial training techniques and adversarial robustness transfer techniques

-