Fast is better than free: Revisiting adversarial training

Цыганов Артем, Адыгамов Ильяс

- Problem of learning robust deep networks remains an active area of research
- Current approaches come at a non-trivial, additional computational cost, often increasing training time by an order of magnitude over standard training

Adversarial training

$$\min_{\theta} \sum_{i} \max_{\delta \in \Delta} \ell(f_\theta(x_i + \delta), y_i).$$

$$\Delta = \{\delta : \|\delta\|_\infty \leq \epsilon\}$$

network f_θ parameterized by θ, a dataset (x_i, y_i), a loss function ℓ and a threat model Δ

Fast Gradient Sign Method

$$X' = X + \epsilon * sign(\nabla_x J(X, ytrue))$$

$$\delta^\star = \epsilon \cdot \text{sign}(\nabla_x \ell(f(x), y)).$$

Projected Gradient Descent adversarial training

Algorithm 1 PGD adversarial training for T epochs, given some radius ϵ, adversarial step size α and N PGD steps and a dataset of size M for a network f_θ

for $t = 1 \ldots T$ **do**
  for $i = 1 \ldots M$ **do**
    // Perform PGD adversarial attack
    $\delta = 0$ *// or randomly initialized*
    for $j = 1 \ldots N$ **do**
      $\delta = \delta + \alpha \cdot \mathrm{sign}(\nabla_\delta \ell(f_\theta(x_i + \delta), y_i))$
      $\delta = \max(\min(\delta, \epsilon), -\epsilon)$
    end for
    $\theta = \theta - \nabla_\theta \ell(f_\theta(x_i + \delta), y_i)$ *// Update model weights with some optimizer, e.g. SGD*
  end for
end for

Free adversarial training

Algorithm 2 "Free" adversarial training for T epochs, given some radius ϵ, N minibatch replays, and a dataset of size M for a network f_θ

$\delta = 0$
// Iterate T/N times to account for minibatch replays and run for T total epochs
for $t = 1 \ldots T/N$ **do**
  for $i = 1 \ldots M$ **do**
    // Perform simultaneous FGSM adversarial attack and model weight updates T times
    for $j = 1 \ldots N$ **do**
      // Compute gradients for perturbation and model weights simultaneously
      $\nabla_\delta, \nabla_\theta = \nabla \ell(f_\theta(x_i + \delta), y_i)$
      $\delta = \delta + \epsilon \cdot \mathrm{sign}(\nabla_\delta)$
      $\delta = \max(\min(\delta, \epsilon), -\epsilon)$
      $\theta = \theta - \nabla_\theta$ *// Update model weights with some optimizer, e.g. SGD*
    end for
  end for
end for

Fast adversarial training

Algorithm 3 FGSM adversarial training for T epochs, given some radius ϵ, N PGD steps, step size α, and a dataset of size M for a network f_θ

for $t = 1 \ldots T$ **do**
  for $i = 1 \ldots M$ **do**
    // Perform FGSM adversarial attack
    $\delta = \text{Uniform}(-\epsilon, \epsilon)$
    $\delta = \delta + \alpha \cdot \text{sign}(\nabla_\delta \ell(f_\theta(x_i + \delta), y_i))$
    $\delta = \max(\min(\delta, \epsilon), -\epsilon)$
    $\theta = \theta - \nabla_\theta \ell(f_\theta(x_i + \delta), y_i)$ *// Update model weights with some optimizer, e.g. SGD*
  end for
end for

Revisiting FGSM adversarial training

- **Очень важен размер шага**
  - **при alpha = 10/255 модель настолько же устойчива, как free**
  - **при alpha = 2eps модель переобучается**
- **Стоимость вычислений**
  - **Отдельно считаем градиенты для perturbation и для весов модели**
  - **Поэтому стоимость вычислений на одной эпохи == стоимости вычислений на двух эпохах при обычном обучении**

Dawnbench improvements

- **Cycling learning rate**
- **Mixed-precision arithmetic**

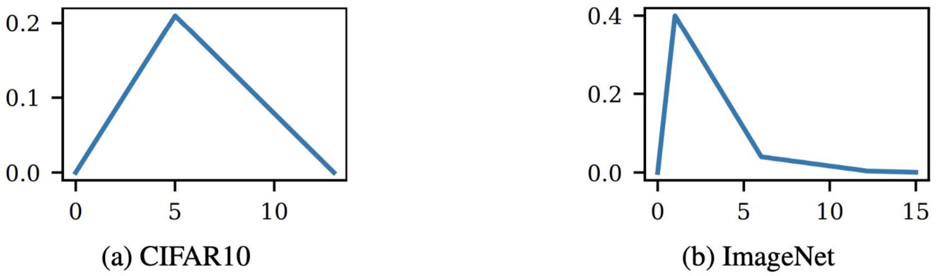

Figure 1: Cyclic learning rates used for FGSM adversarial training on CIFAR10 and ImageNet over epochs. The ImageNet cyclic schedule is decayed further by a factor of 10 in the second and third phases.

Experiments

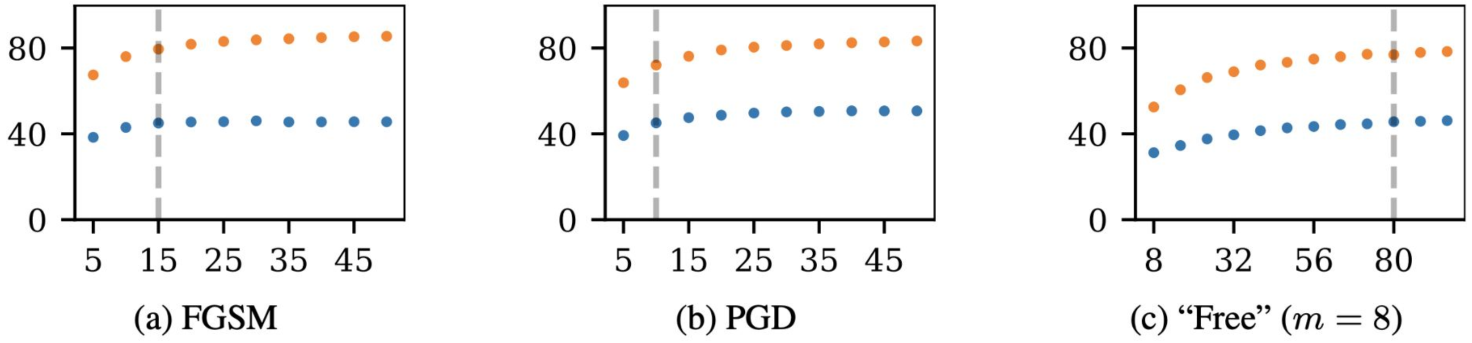

Figure 2: Performance of models trained on CIFAR10 at $\epsilon = 8/255$ with cyclic learning rates and half precision, given varying numbers of epochs across different adversarial training methods. Each point denotes the average model performance over 3 independent runs, where the x axis denotes the number of epochs N the model was trained for, and the y axis denotes the resulting accuracy. The orange dots measure accuracy on natural images and the blue dots plot the empirical robust accuracy on adversarial images. The vertical dotted line indicates the minimum number of epochs needed to train a model to 45% robust accuracy.

Experiments

Table 3: Time to train a robust CIFAR10 classifier to 45% robust accuracy using various adversarial training methods with the DAWNBench techniques of cyclic learning rates and mixed-precision arithmetic, showing significant speedups for all forms of adversarial training.

Method	Epochs	Seconds/epoch	Total time (minutes)
DAWNBench + PGD-7	10	104.94	17.49
DAWNBench + Free ($m = 8$)	80	13.08	17.44
DAWNBench + FGSM	15	25.36	6.34
PGD-7 (Madry et al., 2017)[5]	205	1456.22	4965.71
Free ($m = 8$) (Shafahi et al., 2019)[6]	205	197.77	674.39

Experiments

Table 4: Imagenet classifiers trained with adversarial training methods at $\epsilon = 2/255$ and $\epsilon = 4/255$.

Method	ϵ	Standard acc.	PGD+1 restart	PGD+10 restarts	Total time (hrs)
FGSM	$2/255$	60.90%	43.46%	43.43%	12.14
Free $(m = 4)$	$2/255$	64.37%	43.31%	43.28%	52.20
FGSM	$4/255$	55.45%	30.28%	30.18%	12.14
Free $(m = 4)$	$4/255$	60.42%	31.22%	31.08%	52.20

Table 5: Time to train a robust ImageNet classifier using various fast adversarial training methods

Method	Precision	Epochs	Min/epoch	Total time (hrs)
FGSM (phase 1)	single	6	22.65	2.27
FGSM (phase 2)	single	6	65.97	6.60
FGSM (phase 3)	single	3	114.45	5.72
FGSM	single	15	-	14.59
Free $(m = 4)$	single	92	34.04	52.20
FGSM (phase 1)	mixed	6	20.07	2.01
FGSM (phase 2)	mixed	6	53.39	5.34
FGSM (phase 3)	mixed	3	95.93	4.80
FGSM	mixed	15	-	12.14
Free $(m = 4)$	mixed	92	25.28	38.76

Catastrophic Overfitting

- **неправильная начальная инициализация**
- **неправильный размер шага**
- **ограниченность тренировочной выборки**

Рецензия

Плюсы

- Авторы смогли заставить работать прежде не рабочий метод FGSM(новизна)
- Широкое экспериментальное исследование и впечатляющие результаты(сравнимое качество при очень большом выигрыше во времени)
- Сама идея очень простая
- Статья написана доходчиво, есть даже короткое введение в область

Минусы

- Нет теории(почти)
- Нет сравнений предлагаемого метода с методом проекции градиента при больших масштабах(ImageNet)
- Новизна есть, но разница между существующими методами очень маленькая

Вопросы

- Почему авторы взяли именно эти трюки из DAWNBench и как их отбирали?
- Какое качество дает авторский метод без трюков из DAWNBench?

Итоговая оценка

- **Оценка: 6** (Marginally above the acceptance threshold.)
- **Уверенность: 4** (You are confident in your assessment, but not absolutely certain.)