Less is More: Data Pruning for Faster Adversarial Training

Yize Li[1,†], Pu Zhao[1], Xue Lin[1], Bhavya Kailkhura[2] and Ryan Goldhahn[2]

[1]Northeastern University, 360 Huntington Ave, Boston, MA 02115
[2]Lawrence Livermore National Laboratory, 7000 East Ave, Livermore, CA 94550

Abstract

Deep neural networks (DNNs) are sensitive to adversarial examples, resulting in fragile and unreliable performance in the real world. Although adversarial training (AT) is currently one of the most effective methodologies to robustify DNNs, it is computationally very expensive (*e.g.*, $5 \sim 10\times$ costlier than standard training). To address this challenge, existing approaches focus on single-step AT, referred to as Fast AT, reducing the overhead of adversarial example generation. Unfortunately, these approaches are known to fail against stronger adversaries. To make AT computationally efficient without compromising robustness, this paper takes a different view of the efficient AT problem. Specifically, we propose to minimize redundancies at the data level by leveraging *data pruning*. Extensive experiments demonstrate that the data pruning based AT can achieve similar or superior robust (and clean) accuracy as its unpruned counterparts while being significantly faster. For instance, proposed strategies accelerate CIFAR-10 training up to $3.44\times$ and CIFAR-100 training to $2.02\times$. Additionally, the data pruning methods can readily be reconciled with existing adversarial acceleration tricks to obtain the striking speed-ups of $5.66\times$ and $5.12\times$ on CIFAR-10, $3.67\times$ and $3.07\times$ on CIFAR-100 with TRADES and MART, respectively.

Keywords

Adversarial Robustness, Adversarial Data Pruning, Efficient Adversarial Training

1. Introduction

Deep neural networks (DNNs) achieve great success in various machine learning tasks, such as image classification [1, 2], object detection [3, 4], language modeling [5, 6] and so on. However, the reliability and security concerns of DNNs limit their wide deployment in real-world applications. For example, imperceptible perturbations added to inputs by adversaries (known as adversarial examples) [7, 8, 9] can cause incorrect predictions during inference. Therefore, many research efforts are devoted to designing robust DNNs against adversarial examples [10, 11, 12].

Adversarial Training (AT) [13] is one of the most effective defense approaches to improving adversarial robustness. AT is formulated as a min-max problem, with the inner maximization aiming to generate adversarial examples, and the outer minimization aiming to train a model based on them. However, to achieve better defense with higher robustness, the iterative AT is required to generate stronger adversarial examples with more steps in the inner problem, leading to expensive computation costs. In response to this difficulty, a number of approaches investigate efficient AT, such as Fast AT [14] and their variants [15, 16] via single-step adversarial attacks. Unfortunately, these cheaper training approaches are known to attain poor performance on stronger adversaries and suffer from 'catastrophic overfitting' [14, 17], where Projected Gradient Descent (PGD) robustness is gained at the beginning, but later the robust accuracy decreases to 0 suddenly. In this regard, there does not seem to exist a satisfactory solution to achieve optimal robustness with moderate computation cost.

In this paper, we propose to overcome the above limitation by exploring a new perspective—leveraging ***data pruning*** during AT. Differing from the prior Fast AT-based solutions that focus on the AT algorithm, we attain efficiency by selecting the representative subset of training samples and performing AT on this smaller dataset.

Although several recent works explore data pruning for efficient standard training (see [18] for a survey), data pruning for efficient AT is not well investigated. To the best of our knowledge, the most relevant one is [19], which speeds up AT by the loss-based data pruning. However, the random sub-sampling outperforms their data pruning schemein terms of clean accuracy, robustness, and training efficiency, raising doubts about the feasibility of the proposed approach. In contrast, we propose to perform data pruning in two ways: 1) by maximizing the log-likelihood of the subset on the validation dataset, and 2) by minimizing the gradient disparity between the subset and the full dataset. We implement these approaches with two AT objectives: TRADES [20] and MART [21]. Experimental results show that we can achieve training acceleration up to $3.44\times$ on CIFAR-10 and $2.02\times$ on CIFAR-100. In addition, incorporating our proposed data pruning with Bullet-Train [22], which allocates dynamic computing cost to categorized training data, further im-

The AAAI-23 Workshop on Artificial Intelligence Safety (SafeAI 2023), Feb 13-14, 2023, Washington, D.C., US

† Corresponding author.
li.yize@northeastern.edu (Y. Li); p.zhao@northeastern.edu (P. Zhao); xue.lin@northeastern.edu (X. Lin); kailkhura1@llnl.gov (B. Kailkhura); goldhahn1@llnl.gov (R. Goldhahn)

proves the speed-ups by $5.66\times$ and $3.67\times$ on CIFAR-10 and CIFAR-100, respectively. Our main contributions are summarized below.

- We explore efficient AT from the lens of data pruning, where the acceleration is achieved by only focusing on the representative subset of the data.
- We propose two data pruning algorithms, Adv-GRAD-MATCH and Adv-GLISTER, and perform a comprehensive experimental study. We demonstrate that our data pruning methods yield consistent effectiveness across diverse robustness evaluations, *e.g.*, PGD [13] and AutoAttack [23].
- Furthermore, combining our efficient AT framework with the existing Bullet-Train approach [22] achieves state-of-the-art performance in training cost.

2. Related Work

Adversarial attacks and defenses. Adversarial attacks [13, 24, 25, 26, 27] refer to detrimental techniques that inject imperceptible perturbations into the inputs and mislead decision making process of networks. In this paper, we mainly investigate ℓ_p attacks, where $p \in \{0, 1, 2, \infty\}$. Fast Gradient Sign Method (FGSM) [24] is the cheapest one-shot adversarial attack. Basic Iterative Method (BIM) [28], Projected Gradient Descent (PGD) [13] and CW [25] are stronger attacks that are iterative in nature. Adversarial examples are used for the assessment of model robustness. AutoAttack [23] ensembles multiple attack strategies to perform a fair and reliable evaluation of adversarial robustness.

Various defense methods [29, 30, 31, 32] have been proposed to tackle the vulnerability of DNNs against adversarial examples, while most of the approaches are built over AT, where perturbed inputs are fed to DNNs to learn from adversarial examples. Projected Gradient Descent (PGD) based AT is one of the most popular defense strategies [13], which uses a multi-step adversary. Training only with adversarial samples can lead to a drop in clean accuracy [33]. To improve the trade-off between accuracy and robustness, TRADES [20] and MART [21] compose the training loss with both the natural error term and the robustness regularization term. Curriculum Adversarial Training (CAT) [34] robustifies DNNs by adjusting PGD steps arranging from weak attack strength to strong attack strength, while Friendly Adversarial Training (FAT) [35] performs early-stopped PGD for adversarial examples.

Efficient adversarial training. Despite PGD-based training showing empirical robustness against adversarial examples, the learning overhead is usually dramatically larger than the standard training, *e.g.*, $5 \sim 10\times$ computation consumption depending on the number of steps used in generating adversarial examples. The major work to achieve training efficiency focuses on how to reduce the number of attack steps and maintain the stability of one-step FGSM-based AT. Free AT [36] performs FGSM perturbations and updates model weights on the simultaneous mini-batch. FAST AT [14] generates FGSM attacks with random initialization but still suffers from 'catastrophic overfitting'. Therefore, Gradient alignment regularization [17], suitable inner interval (step size) for the adversarial direction [16], and Fast Bi-level AT (FAST-BAT) [37] are proposed to prevent such failure.

Data pruning. Efficient learning through data subset selection economizes on training resources. Proxy functions [38, 39] take advantage of the feature representation from the tiny proxy model to select the most informative subset for training the larger one. Coreset-based algorithms [40] mine for a small representative subset that approximates the entire dataset following established criteria. CRAIG [41] selects the training data subset which approximates the full gradient and GRAD-MATCH [42] minimizes the gradient matching error. GLISTER [43] prunes the training data by maximizing log-likelihood for the validation set.

3. Data Pruning Based Adversarial Training

3.1. Preliminaries

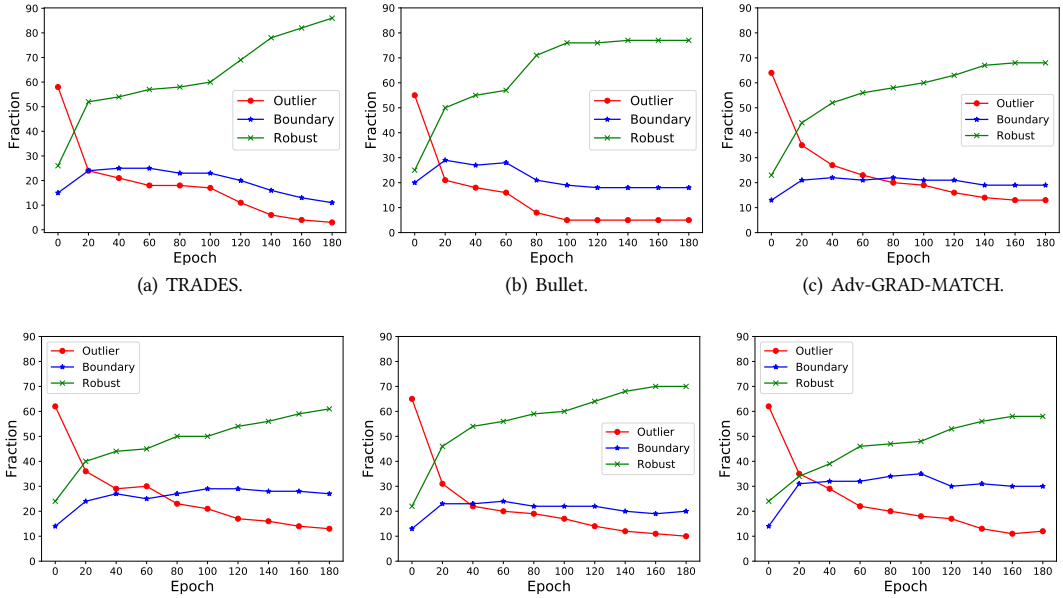
AT [13] aims to solve the min-max optimization problem as follows:

$$\min_{\theta} \frac{1}{|D|} \sum_{(x,y)\in\mathcal{D}} \left[\max_{\delta\in\triangle} \mathcal{L}(\theta; x+\delta, y) \right], \qquad (1)$$

where θ is the model parameter, x and y denote the data sample and label from the training dataset $\mathcal{D}$, δ denotes imperceptible adversarial perturbations injected into x under the norm constraint by the constant strength ϵ, *i.e.*, $\triangle := \{\|\delta\|_\infty \leq \epsilon\}$, and $\mathcal{L}$ is the training loss. During the adversarial procedure, the optimization first maximizes the inner approximation for adversarial attacks and then minimizes the outer training error over the model parameter θ. A typical adversarial example generation procedure involves multiple steps for the stronger adversary, *e.g.*,

$$x^{t+1} = \mathrm{Proj}_{\triangle}\left(x^t + \alpha\, \mathrm{sign}\left(\nabla_{x^t}\mathcal{L}\left(\theta; x^t, y\right)\right)\right), \quad (2)$$

where the projection follows ϵ-ball at the step t with step size α, using the sign of gradients.



(a) TRADES. (b) Bullet. (c) Adv-GRAD-MATCH.

(d) Adv-GLISTER. (e) Adv-GRAD-MATCH&Bullet. (f) Adv-GLISTER&Bullet.

Figure 1: Tracking of adversarial robustness during 200 epochs of training. Red, Green and Blue denote outlier, robust and boundary examples, respectively.

3.2. General Formulation for Adversarial Data Pruning

Our adversarial data pruning consists of two steps: adversarial subset selection and AT with the subset of data. In the specified epoch, adversarial subset selection first finds a representative subset of data from the entire training dataset. Next, AT is performed with the selected subset. Though the size of the subset keeps the same in different iterations, the data in the subset is updated in each iteration based on the different status of the model weights. We formulate the AT with the data subset in Eq. (3) and adversarial subset selection in Eq. (4).

$$\min_{\theta} \frac{1}{k} \sum_{(x,y)\in\mathcal{S}} \left[\max_{\delta\in\triangle} \mathcal{L}(\theta; x+\delta, y) \right], \tag{3}$$

$$\min_{\mathcal{S}\subseteq\mathcal{D},|\mathcal{S}|=k} G(\mathcal{S}) \tag{4}$$

where $\mathcal{D}$ represents the complete training set and δ represents the perturbation under l_∞ norm constraint $\triangle$. The selected subset $\mathcal{S}$ with the size k is obtained by optimizing the function G, which aims to narrow the difference between $\mathcal{D}$ and $\mathcal{S}$ under specific criteria with model parameters θ. Note that the data selection step is performed periodically to achieve computational savings.

Recent data subset selection schemes, GRAD-MATCH [42] and GLISTER [43], have made significant contributions towards efficiently achieving high clean accuracy. We extend these approaches in the context of adversarial robustness. Motivated by GLISTER [43], we first consider training a subset that obtains the optimal adversarial log-likelihood on the validation set in Eq. (5), defined as Adv-GLISTER:

$$G(\mathcal{S}) = \sum_{(x_V,y_V)\in\mathcal{V}} L_V(\theta_S; x_V + \delta_V^*, y_V) \tag{5}$$

where L_V is the negative log-likelihood on validation set; δ_V^* is the adversarial perturbation obtained by maximizing $L_V(\theta_S; x_V + \delta_V, y_V)$.

Another adversarial data pruning approach is inspired by GRAD-MATCH [42], which aims to find the data subset whose gradients closely match those of the full training data. Adv-GRAD-MATCH is formulated as Eq. (6):

$$\begin{aligned} G(\mathcal{S}) = \| &\sum_{(x_S,y_S)\in\mathcal{S}} w\nabla_\theta \mathcal{L}_{\mathcal{S}}(\theta; x_S + \delta_S^*, y_S) \\ &-\nabla_\theta \sum_{(x_D,y_D)\in\mathcal{D}} \mathcal{L}_{\mathcal{D}}(\theta; x_D + \delta_D^*, y_D)\| \end{aligned} \tag{6}$$

where w is the weight vector associated with each instance x_S in the subset $\mathcal{S}$; $\mathcal{L}_S$ and $\mathcal{L}_D$ denote the training loss over the subset and entire dataset; δ_S^* and δ_D^* are adversarial examples obtained by maximizing $L_S(\theta; x_S + \delta_S, y_S)$ and $L_D(\theta; x_D + \delta_D, y_D)$, respectively. During the data selection, the adversarial gradient difference between the weighted subset loss and

Table 1
TRADES results where data pruning methods use only 30% data points on CIFAR-10 and 50% data points on CIFAR-100 for 100 epochs of training.

Dataset	Method	Clean	PGD			AutoAttack	Time/epoch (Speed-up)
			$4/255$	$8/255$	$16/255$		
CIFAR-10	TRADES [20]	82.73	69.17	51.83	19.43	49.06	416.20 (-)
	Bullet [22]	84.60	70.24	50.82	16.05	47.93	193.06 (2.16×)
	Adv-GLISTER (Ours)	77.62	63.06	46.06	16.52	41.61	120.70 (3.45×)
	Adv-GRAD-MATCH (Ours)	75.67	61.85	45.96	17.49	42.19	138.19 (3.01×)
	Adv-GLISTER&Bullet (Ours)	79.21	63.02	44.52	13.33	40.77	**72.91 (5.66×)**
	Adv-GRAD-MATCH&Bullet (Ours)	77.57	62.00	45.13	14.65	41.94	87.38 (4.76×)
CIFAR-100	TRADES [20]	55.85	40.31	27.35	10.71	23.39	387.72 (-)
	Bullet [22]	59.43	42.23	28.08	9.40	23.85	173.59 (2.23×)
	Adv-GLISTER (Ours)	51.26	37.16	24.78	9.49	20.57	202.7 (1.91×)
	Adv-GRAD-MATCH (Ours)	51.03	37.17	24.60	9.70	20.42	206.05 (1.88×)
	Adv-GLISTER&Bullet (Ours)	53.54	37.24	23.91	7.69	20.02	**105.66 (3.67×)**
	Adv-GRAD-MATCH&Bullet (Ours)	52.98	36.92	24.24	8.01	20.17	105.61 (3.67×)

Table 2
MART results where data pruning methods use only 30% data points on CIFAR-10 and 50% data points on CIFAR-100 for 100 epochs of training.

Dataset	Method	Clean	PGD			AutoAttack	Time/epoch (Speed-up)
			$4/255$	$8/255$	$16/255$		
CIFAR-10	MART [21]	80.96	68.21	52.59	19.52	46.94	329.54 (-)
	Bullet [22]	85.29	70.92	50.64	13.33	43.77	199.42 (1.65×)
	Adv-GLISTER (Ours)	71.97	60.13	46.25	16.59	39.86	95.68 (3.44×)
	Adv-GRAD-MATCH (Ours)	73.67	61.35	47.07	18.16	40.98	106.51 (3.09×)
	Adv-GLISTER&Bullet (Ours)	73.87	59.89	44.01	14.20	38.99	**64.31 (5.12×)**
	Adv-GRAD-MATCH&Bullet (Ours)	78.78	64.42	46.72	13.50	39.53	77.11 (4.27×)
CIFAR-100	MART [21]	54.85	39.24	25.08	8.59	22.66	307.43 (-)
	Bullet [22]	57.44	39.22	24.14	6.66	21.55	187.73 (1.64×)
	Adv-GLISTER (Ours)	46.36	34.37	24.01	9.20	19.79	152.11 (2.02×)
	Adv-GRAD-MATCH (Ours)	48.07	36.19	26.11	10.79	21.24	153.86 (2.00×)
	Adv-GLISTER&Bullet (Ours)	52.13	35.07	20.67	5.64	18.21	**100.22 (3.07×)**
	Adv-GRAD-MATCH&Bullet (Ours)	52.46	35.81	22.20	6.48	18.68	113.03 (2.72×)

the complete dataset loss is minimized so as to produce the optimum subset and corresponding weights.

4. Experiments

4.1. Experiment Setup

To evaluate the efficiency and generality of the proposed method, we apply adversarial training loss functions from TRADES [20] or MART [21] on the standard datasets, CIFAR-10, CIFAR-100 [44] trained on ResNet-18 [45]. Our adversarial data pruning methods include Adv-GRAD-MATCH and Adv-GLISTER with different data portions (subset size) $[30\%, 50\%]$ with 100 and 200 epochs where the selection interval is 20 (i.e., perform adversarial subset selection every 20 epochs of AT). The original training dataset is divided into the train (90%) and the validation set (10%) in Adv-GLISTER. The optimizer is SGD with momentum 0.9 and weight decay 2e-4 for TRADES and 3.5e-3 for MART. For Adv-GRAD-MATCH and Adv-GLISTER, the initial learning rate is 0.01 and 0.02 on CIFAR-10 and 0.08 and 0.05 on CIFAR-100 respectively. Besides the original TRADES [20] and MART [21] methods, we also compare our approach with Bullet-Train [22]. PGD attack [13] (PGD-50-10) is adopted for evaluating the robust accuracy, ranging from low magnitude ($\epsilon = 4/255$) to high magnitude ($\epsilon = 16/255$) with 50 iterations as well as 10 restarts at the step-size $\alpha = 2/255$ under l_∞-norm. Moreover, AutoAttack [23] is leveraged for the reliable robustness evaluation. Additionally, our methods can also be combined with Bullet-Train [22] and we term them as Adv-GRAD-MATCH&Bullet and Adv-GLISTER&Bullet.

4.2. Main Results

Table 1 shows the results of our Adv-GLISTER and Adv-GRAD-MATCH for TRADES compared with the original TRADES and Bullet-Train methods. The compar-

Table 3
100 *v.s.* 200 epoch TRADES CIFAR-10 results with ResNet-18 when using 30% data points with robustness regularization factor to be 1.

Method	Epoch	Clean	PGD			AutoAttack
			$4/255$	$8/255$	$16/255$	
Adv-GLISTER	100	77.62	63.06	46.06	16.52	41.61
Adv-GRAD-MATCH	100	75.61	60.81	45.76	17.49	42.19
Adv-GLISTER	200	78.76	64.15	46.11	16.92	42.43
Adv-GRAD-MATCH	200	75.75	61.24	46.49	18.55	43.63

Table 4
TRADES results on CIFAR-10 with ResNet-18 using 30% data samples under different selection counts for 200 epoch training.

Method	Number of selections	Clean	PGD			AutoAttack	Speed-up
			$4/255$	$8/255$	$16/255$		
TRADES	-	83.32	68.91	49.64	17.31	47.53	-
Adv-GLISTER	4	75.80	60.48	44.62	16.07	40.44	3.15×
Adv-GRAD-MATCH	4	73.80	60.43	46.06	18.33	43.03	2.83×
Adv-GLISTER	9	78.76	64.15	46.11	16.92	42.43	2.93×
Adv-GRAD-MATCH	9	75.75	61.24	46.49	18.55	43.63	2.75×

ison is in terms of clean and robust accuracy (under two attack methods, PGD Attack [13] and AutoAttack [23]) along with the training speed-up. We observe that compared to the baselines, the training efficiency of our method is improved significantly on CIFAR-10, while the decrease happens on the clean accuracy and robustness under AutoAttack and PGD attacks for different values of ϵ. Especially, for $\epsilon = 16/255$, the robust accuracy can be improved from 16.05% (Bullet-Train [22]) to 16.52% and 17.49% with our Adv-GRAD-MATCH and Adv-GLISTER, indicating our defensive capability on powerful attacks. As displayed in Table 1, our Adv-GRAD-MATCH and Adv-GLISTER reduce the training overheads (seconds per epoch) enormously and achieve $3.44\times$ and $3.09\times$ training speed-ups. After combining our approaches with Bullet-Train [22], an even faster acceleration of $5.12\times$ can be reached.

On CIFAR-100, the validity of our schemes is consistent as well. The reason why both clean and robust accuracy drop might be that our data pruning schemes struggle with the dimensionality and complexity of the dataset. Regardless, our schemes still result in conspicuous computation savings compared with other baselines.

To understand the robustness improvements of our schemes, we track the dynamics of the outlier, robust, and boundary sets (similar to [22]) using PGD-5-1 attack. Without any attack, the outlier examples have already been mistaken by the model, but boundary and robust examples are correctly identified. After adversarial attacks, boundary examples are incorrectly classified while robust examples are still correctly classified. Fig. 1 displays the dynamics of the outlier, boundary, and robust examples on CIFAR-10 for various schemes. During the model training and data selection, the number of robust samples gradually increases and eventually dominates, while the number of outliers and boundary data points decreases over epochs, revealing similar achievements in TRADES-based AT and data pruning-based methods. In addition, the ultimate portions of three sets explain the clean accuracy and robustness degrading of our approaches. In detail, two baselines obtain more robust samples and fewer boundary and outlier examples.

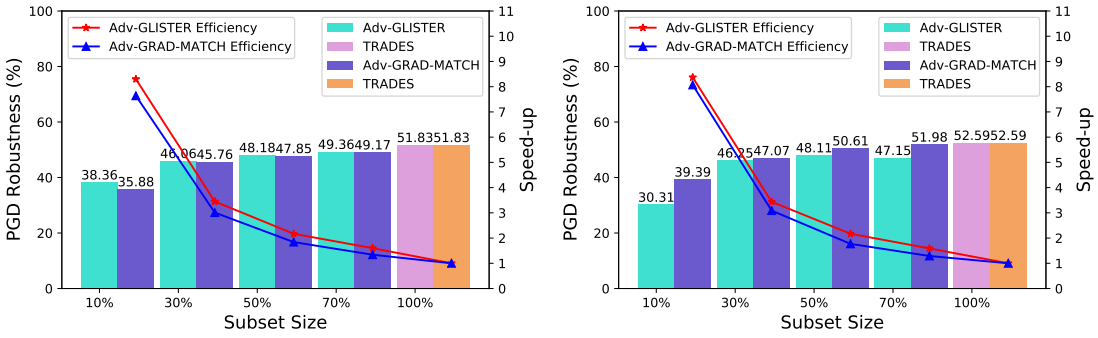
We further evaluate the performances of adversarial data pruning based on the loss of MART in Table 2. Results are consistent with our findings on TRADES in Table 1.

4.3. Ablation Studies

Epoch. We first consider the training epoch. Table 3 shows that longer training improves both clean and robust accuracy. Due to the shrinking data size, more epochs are required to enhance data-efficient adversarial learning, in alignment with standard data pruning training. However, 100-epoch training appears to be sufficient for the small dataset.

Subset Size. We experiment with different subset sizes. Moving from the extremely small subset (10% of the full training set) to a larger subset (70%) in Fig. 2, the observation is that robust accuracy gradually increases to that of the full dataset. This highlights the benefit of pruning with optimal subset size. We can see that 30% is an appropriate choice for the CIFAR-10 subset size, after taking the global efficiency into account.

Number of selection rounds. In Sec. 4.2, our experiments perform adversarial data pruning every 20 epochs (with 9 selections). Here we present the results of data pruning every 40 epochs (with 4 selections). As shown in



(a) TRADES. (b) MART.

Figure 2: PGD evaluation ($\epsilon = 8/255$) with the corresponding speed-up under different subset sizes for 100 epoch CIFAR-10 training. Note that when the size is 100%, data pruning methods are not applied and the speed-up is compared with the baselines (TRADES or MART).

Table 4, 9 selections can achieve better clean and robust accuracy with comparable acceleration.

5. Conclusion and Future Work

In this paper, we investigated efficient adversarial training from a data-pruning perspective. With comprehensive experiments, we demonstrated that proposed adversarial data pruning approaches outperform the existing baselines by mitigating substantial computational overhead. These positive results pave a path for future research on accelerating AT by minimizing redundancy at the data level. Our future work will focus on designing more accurate pruning schemes for large-scale datasets.

Acknowledgment

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 and was supported by LLNL-LDRD Program under Project No. 20-SI-005 (LLNL-CONF-842760).

References

[1] Q. Xie, M.-T. Luong, E. Hovy, Q. V. Le, Self-training with noisy student improves imagenet classification, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[2] P. Foret, A. Kleiner, H. Mobahi, B. Neyshabur, Sharpness-aware minimization for efficiently improving generalization, in: International Conference on Learning Representations (ICLR), 2021.

[3] Z.-Q. Zhao, P. Zheng, S.-T. Xu, X. Wu, Object detection with deep learning: A review, IEEE Transactions on Neural Networks and Learning Systems 30 (2019) 3212–3232. doi:10.1109/TNNLS.2018.2876865.

[4] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, B. Lee, A survey of modern deep learning based object detection models, Digital Signal Processing 126 (2022) 103514.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems (NeurIPS), 2017.

[6] S. Borgeaud, A. Mensch, J. Hoffmann, T. Cai, E. Rutherford, K. Millican, G. B. Van Den Driessche, J.-B. Lespiau, B. Damoc, A. Clark, D. De Las Casas, A. Guy, J. Menick, R. Ring, T. Hennigan, S. Huang, L. Maggiore, C. Jones, A. Cassirer, A. Brock, M. Paganini, G. Irving, O. Vinyals, S. Osindero, K. Simonyan, J. Rae, E. Elsen, L. Sifre, Improving language models by retrieving from trillions of tokens, in: Proceedings of the 39th International Conference on Machine Learning (ICML), 2022.

[7] P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi, C.-J. Hsieh, Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models, in: Proceedings of the ACM Workshop on Artificial Intelligence and Security, ACM, 2017.

[8] C. Xiao, B. Li, J. yan Zhu, W. He, M. Liu, D. Song, Generating adversarial examples with adversarial networks, in: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence(IJCAI), 2018.

[9] F. Tramer, N. Carlini, W. Brendel, A. Madry, On adaptive attacks to adversarial example defenses,

in: Advances in Neural Information Processing Systems (NeurIPS), 2020.

[10] A. Athalye, N. Carlini, D. Wagner, Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples, in: Proceedings of the 35th International Conference on Machine Learning (ICML), 2018.

[11] E. Wong, Z. Kolter, Provable defenses against adversarial examples via the convex outer adversarial polytope, in: Proceedings of the 35th International Conference on Machine Learning (ICML), 2018.

[12] H. Salman, J. Li, I. Razenshteyn, P. Zhang, H. Zhang, S. Bubeck, G. Yang, Provably robust deep learning via adversarially trained smoothed classifiers, in: Advances in Neural Information Processing Systems (NeurIPS), 2019.

[13] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, A. Vladu, Towards deep learning models resistant to adversarial attacks, in: International Conference on Learning Representations (ICLR), 2018.

[14] E. Wong, L. Rice, J. Z. Kolter, Fast is better than free: Revisiting adversarial training, in: International Conference on Learning Representations (ICLR), 2020.

[15] B. S. Vivek, R. Venkatesh Babu, Single-step adversarial training with dropout scheduling, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[16] H. Kim, W. Lee, J. Lee, Understanding catastrophic overfitting in single-step adversarial training, in: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), volume 35, 2021, pp. 8119–8127.

[17] M. Andriushchenko, N. Flammarion, Understanding and improving fast adversarial training, in: Advances in Neural Information Processing Systems (NeurIPS), 2020.

[18] B. R. Bartoldson, B. Kailkhura, D. Blalock, Compute-efficient deep learning: Algorithmic trends and opportunities, arXiv preprint arXiv:2210.06640 (2022).

[19] M. Kaufmann, Y. Zhao, I. Shumailov, R. Mullins, N. Papernot, Efficient adversarial training with data pruning, in: arXiv, 2022.

[20] H. Zhang, Y. Yu, J. Jiao, E. P. Xing, L. E. Ghaoui, M. I. Jordan, Theoretically principled trade-off between robustness and accuracy, in: International Conference on Machine Learning (ICML), 2019.

[21] Y. Wang, D. Zou, J. Yi, J. Bailey, X. Ma, Q. Gu, Improving adversarial robustness requires revisiting misclassified examples, in: International Conference on Learning Representations (ICLR), 2020.

[22] W. Hua, Y. Zhang, C. Guo, Z. Zhang, G. E. Suh, Bullettrain: Accelerating robust neural network training via boundary example mining, in: Advances in Neural Information Processing Systems (NeurIPS), 2021.

[23] F. Croce, M. Hein, Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks, in: International Conference on Machine Learning (ICML), 2020.

[24] I. J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, in: arXiv, 2015.

[25] N. Carlini, D. Wagner, Towards evaluating the robustness of neural networks, in: IEEE Symposium on Security and Privacy (S&P), IEEE, 2017.

[26] F. Croce, M. Hein, Sparse and imperceivable adversarial attacks, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019.

[27] Q. Zhang, X. Li, Y. Chen, J. Song, L. Gao, Y. He, H. Xue, Beyond imagenet attack: Towards crafting adversarial examples for black-box domains, in: International Conference on Learning Representations (ICLR), 2022.

[28] A. Kurakin, I. Goodfellow, S. Bengio, Adversarial examples in the physical world, 2016. URL: https://arxiv.org/abs/1607.02533. doi:10.48550/ARXIV.1607.02533.

[29] D. Meng, H. Chen, Magnet: A two-pronged defense against adversarial examples, in: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, 2017.

[30] F. Liao, M. Liang, Y. Dong, T. Pang, X. Hu, J. Zhu, Defense against adversarial attacks using high-level representation guided denoiser, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[31] A. Mustafa, S. Khan, M. Hayat, R. Goecke, J. Shen, L. Shao, Adversarial defense by restricting the hidden space of deep neural networks, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019.

[32] Y. Gong, Y. Yao, Y. Li, Y. Zhang, X. Liu, X. Lin, S. Liu, Reverse engineering of imperceptible adversarial image perturbations, in: International Conference on Learning Representations (ICLR), 2022.

[33] D. Su, H. Zhang, H. Chen, J. Yi, P.-Y. Chen, Y. Gao, Is robustness the cost of accuracy? – a comprehensive study on the robustness of 18 deep image classification models, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018.

[34] Q.-Z. Cai, C. Liu, D. Song, Curriculum adversarial training, in: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18, International Joint Conferences on Artificial Intelligence Organization, 2018, pp. 3740–3747. URL: https://doi.org/10.24963/ijcai.2018/520. doi:10.24963/ijcai.2018/520.

[35] J. Zhang, X. Xu, B. Han, G. Niu, L. Cui, M. Sugiyama,

M. Kankanhalli, Attacks which do not kill training make adversarial learning stronger, in: Proceedings of the 37th International Conference on Machine Learning (ICML), 2020.

[36] A. Shafahi, M. Najibi, M. A. Ghiasi, Z. Xu, J. Dickerson, C. Studer, L. S. Davis, G. Taylor, T. Goldstein, Adversarial training for free!, in: Advances in Neural Information Processing Systems (NeurIPS), 2019.

[37] Y. Zhang, G. Zhang, P. Khanduri, M. Hong, S. Chang, S. Liu, Revisiting and advancing fast adversarial training through the lens of bi-level optimization, in: International Conference on Machine Learning (ICML), 2022.

[38] C. Coleman, C. Yeh, S. Mussmann, B. Mirzasoleiman, P. Bailis, P. Liang, J. Leskovec, M. Zaharia, Selection via proxy: Efficient data selection for deep learning, in: International Conference on Learning Representations (ICLR), 2020. URL: https://openreview.net/forum?id=HJg2b0VYDr.

[39] V. Kaushal, R. Iyer, S. Kothawade, R. Mahadev, K. Doctor, G. Ramakrishnan, Learning from less data: A unified data subset selection and active learning framework for computer vision, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2019.

[40] D. Feldman, Core-Sets: Updated Survey, Springer International Publishing, Cham, 2020, pp. 23–44. URL: https://doi.org/10.1007/978-3-030-29349-9_2. doi:10.1007/978-3-030-29349-9_2.

[41] B. Mirzasoleiman, J. Bilmes, J. Leskovec, Coresets for data-efficient training of machine learning models, in: Proceedings of the 37th International Conference on Machine Learning (ICML), 2020.

[42] K. Killamsetty, D. S, G. Ramakrishnan, A. De, R. Iyer, Grad-match: Gradient matching based data subset selection for efficient deep model training, in: Proceedings of the 38th International Conference on Machine Learning (ICML), 2021.

[43] K. Killamsetty, D. Sivasubramanian, G. Ramakrishnan, R. Iyer, Glister: Generalization based data subset selection for efficient and robust learning, in: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), volume 35, 2021, pp. 8110–8118.

[44] A. Krizhevsky, G. Hinton, Learning multiple layers of features from tiny images, Master's thesis, Department of Computer Science, University of Toronto (2009).

[45] K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks, in: European conference on computer vision (ECCV), Springer, 2016, pp. 630–645.