Test-time Adaptation of Residual Blocks against Poisoning and Backdoor Attacks

Arnav Gudibande
UC Berkeley
arnavg@berkeley.edu

Xinyun Chen
UC Berkeley
xinyun.chen@berkeley.edu

Yang Bai
Tsinghua University
y-bai17@mails.tsinghua.edu.cn

Jason Xiong
UC Berkeley
xjs01@berkeley.edu

Dawn Song
UC Berkeley
dawnsong@cs.berkeley.edu

Abstract

Data poisoning has become a major security threat for deep neural networks, where the attacker injects maliciously crafted poisoning samples into the training set to mislead the model prediction. Numerous poisoning attack strategies have been proposed recently, which are mostly able to alter the model behavior or embed backdoors with a small number of poisoning samples. Current defenses for these attacks either do not generalize to diverse threat models or suffer from a huge computational cost. In this work, we propose `ReScaler`, a parameter-efficient defense that adapts the residual blocks at the test time. Specifically, `ReScaler` learns a scalar for each residual connection to downweight potentially redundant non-linear transformations, in favor of the features propagated through the skip connections. Our evaluation on several state-of-the-art poisoning attacks and different residual networks shows that `ReScaler` effectively defends against different attack algorithms, without introducing significant computational overhead. Our test-time adaptation scheme introduces a novel way of approaching defenses for poisoning and backdoor attacks, and also brings up broader questions about the connection between the architectural design and the vulnerability against attacks. [1]

1. Introduction

In data poisoning, an attacker can inject poisoning samples in such a way as to degrade model performance or embed a backdoor [1, 9, 22]. Typically, these training-time attacks can be successfully launched with a small number of poisoning samples. Furthermore, recent works on clean-label data poisoning demonstrate that the poisoning samples can be hardly recognized even with manual investigation [20]. As a result, these vulnerabilities can be especially severe for models deployed into safety-critical environments, such as auto-driving or facial recognition.

Existing defenses for data poisoning [4, 6, 8, 17, 23, 26, 29] largely suffer from limited generalization to different attack algorithms, large computational costs, or considerable drop in the prediction performance [15]. Facing these limitations, we aim to propose an effective defense for data poisoning and backdoor attacks that can effectively balance these concerns of generalization, model performance, and computational expenses.

In this work, we propose `ReScaler`, a test-time adaptation of the residual block to defend against poisoning and backdoor attacks. Our defense is motivated by prior studies of residual networks, which discuss how different residual blocks in the model architecture contribute to its predictions. Specifically, Veit et al. [25] and Greff et. al. [10] demonstrate that residual networks can be interpreted as stacked ensembles consisting of multiple residual pathways, and as a result are surprisingly resilient to subtle modifications at the test time. Zhang et al. [31] further show the existence of critical layers in residual networks, and demonstrate how test-time changes like re-initialization or reordering of residual blocks can moderately defend against some adversarial attacks.

Based on this work by Veit et al. and Zhang et al., we hypothesize that in a poisoned model, there are particular pathways that are strongly responsible for the ultimate prediction of a target image. In order to counteract this, `ReScaler` learns to downweight the features of certain residual blocks during test-time. Specifically, we introduce trainable scalar parameters to control the weight of features through each convolution layer in a residual block, and count more on features propagated through the skip connection whenever appropriate. Given the small number of additional parameters introduced in the `ReScaler`, we can efficiently update these scalar parameters at test time,

[1] We will release the code upon publication.

without modifying other model parameters. Therefore, our defense can be directly applied to any pre-trained residual network to reverse the effect of attacks.

We evaluate ReScaler on a number of poisoning and backdoor attacks, especially the clean-label attacks on which most prior defenses fail [17]. ReScaler effectively defends against the attacks at the inference time, while maintaining a high accuracy on clean samples. Moreover, ReScaler is computationally efficient, enables us to identify the critical features for learning poisoning samples, and provides a different view to understand the transferability of different attack schemes.

2. Poisoning and Backdoor Attacks

In poisoning and backdoor attacks [1, 9, 14, 20, 22, 24], the attacker injects a small fraction of poisoning samples into the (clean) training data and thus induces some abnormal model behavior, revealing the model vulnerability in the training phase. *Data poisoning* aims to induce one model to give wrong predictions on some specific target images (without any trigger) during test time. When embedding with a *backdoor attack*, attackers further manipulate the model during test-time by applying specific triggers onto input samples.

Notations. For data poisoning, the specific target images during inference are denoted as $\boldsymbol{x}_t$, whose ground truth labels are dubbed as the *target class* y_t and the (poisoning) predictive labels are dubbed as *poison/adversarial class* y_{adv}. For backdoor attacks, the target images $\boldsymbol{x}_t$ added with the specific triggers Δ will be misled at the test time. We briefly introduce a series of poisoning and backdoor attacks adopted in this paper.

Feature Collision (FC). Shafahi et al. [22] optimize the poisoning samples by restricting their feature representations to lie close to that of target image while maintaining the visual similarity with correspondingly base images.

Convex/Bullseye Polytope (CP/BP). Zhu et al. [33] (CP) propose to surround target images in feature space. They express the feature of target image as a convex combination of features extracted from poisoning samples. Then Aghakhani et al. [1] (BP) step further by forcing the feature of target image to locate at the mean/center of such convex poisoning features.

Clean-label Backdoors. Turner et al. [24] generate poisoning samples by only adding the adversarial perturbations as triggers, dubbed as Clean Label Backdoor Attack (CLBD). Inspired by Feature Collision (FC), Saha et al. [20] propose the Hidden Trigger Backdoor Attack (HTBD) by optimizing and hiding the patch-wise triggers in feature-level to make them invisible.

3. ReScaler: Test-time Adaptation of Residual Blocks

In this section, we present ReScaler as a method to defend against poisoning and backdoor attacks and describe the test-time adaptation algorithm.

3.1. ReScaler Formulation

Deep neural networks have shown a strong learning capability with a huge number of parameters. In order to alleviate the vanishing gradients with the increase of the network depth, the residual module was proposed and has been widely applied, which uses an identity shortcut to improve the information flow during forward and backward propagation [12]. Since then, the skip-connection structure has become an important component for state-of-the-art models, including ResNet [12], DenseNet [13] and ResNeXt [28]. Specifically, given an input $\boldsymbol{x}_i$ for the i-th residual block f_i, the output of the block $\boldsymbol{x}_{i+1}$ is

$$\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + f_i(\boldsymbol{x}_i). \tag{1}$$

In ReScaler, for each convolution operation f_{ij} in the i-th block, we learn a scalar parameter w_{ij} to downweight its output as follows:

$$\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + \sum_j (1 - w_{ij}) \cdot f_{ij}(\boldsymbol{x_i}), \tag{2}$$

where the learnable scalar parameters $w_{ij} \in [0, 1]$. When $w_{ij} = 0$, the output of the residual block remains the same. When $w_{ij} = 1$, the output of the convolution operations are ignored, equivalently becoming the identity function. Motivated by prior work that discusses the existence of critical residual blocks that cannot be altered during test time [31], we do not learn scalar parameters for the first block of each residual layer, which are found to be critical to ensure the high prediction performance.

When we apply ReScaler to the ResNet-34 architecture, there are 24 learnable scalar parameters w_{ij} in total. The detailed structure of ReScaler applied onto residual blocks for ResNet-34 is shown in Figure 1.

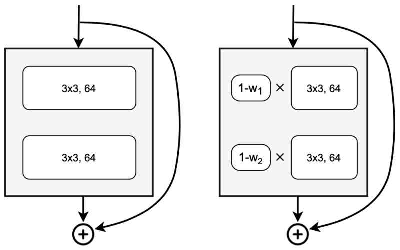

Figure 1. An example of ReScaler design. Left: the residual block in the first layer of ResNet-34. Right: ReScaler applied onto ResNet-34.

3.2. Test-time Adaptation

The lightweight parameterization of ReScaler allows us to efficiently learn the scalar parameters at test time. Before the test-time adaptation, we first initialize all parameters ReScaler w_{ij} to be 0, thus each ReScaler is equivalent to its corresponding original residual block. Given the full parameters of a potentially poisoned model F_θ, an input image $\boldsymbol{x}$ with its predicted output score $F_\theta(\boldsymbol{x})$ and predicted label $y_p(\boldsymbol{x})$, we employ a 1-gradient step update as follows:

$$w_{ij} = w_{ij} + \alpha * \nabla_{w_{ij}} \mathcal{L}_\theta(F_\theta(\boldsymbol{x}), y_p(\boldsymbol{x})), \qquad (3)$$

where α is the step size, and $\mathcal{L}_\theta(F_\theta(\boldsymbol{x}), y_p(\boldsymbol{x}))$ is the cross entropy loss between predicted output score $F_\theta(\boldsymbol{x})$ and predicted label $y_p(\boldsymbol{x})$.

We apply this gradient update for each input image $\boldsymbol{x}$ at test time. To prevent the benign images from being wrongly classified, we introduce an upper bound ϵ on each w_{ij}, so that $w_{ij} \in [0, \epsilon]$.

4. Experiments

In this section, we present the evaluation of ReScaler on various poisoning and backdoor attacks. We first describe our experimental setup, then we discuss the results.

4.1. Experimental Setup

Datasets. We use the CIFAR-10 dataset [16] for evaluation, where many poisoning and backdoor attacks have been successfully launched. For all attacks, poisoning samples are sourced from the training set, and target images are from the validation set. When calculating the validation accuracy of poisoned models, target images are excluded.

Model Architectures. We apply ReScaler onto a number of residual networks, i.e., ResNet-34, ResNet-50, and ResNeXt-29.

Attacks. We evaluate ReScaler against several poisoning and backdoor attacks, specifically FC, CP, and BP, and clean-label backdoor attacks, in particular CLBD and HTBD. We use the same set of attack goals for all clean-label poisoning attacks as evaluated in [21], where each attack goal consists of a pair of randomly sampled target image, target class and poison class. We generate a set of poisoning samples using the attack algorithm under consideration for each attack goal, and we generate 100 poisoning sample sets for each clean-label poisoning attack. We consider an attack to be successful if the model predicts the poison class for the target image. For evaluation of our defense, we only include the successfully poisoned models, thereby the initial attack success rate (ASR) across all models is 100% for each attack.

Metrics. We compute the following metrics to evaluate the effectiveness of the defense:

- *Defended attack success rate*: The percentage of successful attacks after applying the defense. As stated above, the initial attack success rates across all models are 100% for each attack. The lower the defended attack success rate, the better the defense.
- *Put back rate*: The percentage of defended models with their predicted labels of the target images returning to the intended ground-truth labels. The higher the put back rate, the better the defense.
- *Poison confidence*: the average prediction confidence of the target image in the poison class. The lower the poison confidence, the better the defense.
- *Target confidence*: the average confidence of the target image in its ground truth class. The higher the target confidence, the better the defense.

Besides the above metrics, which measure the trade-off between the defense effectiveness and the model performance on clean data, we also measure the average validation accuracy of successfully poisoned models before and after applying the defense.

4.2. Experimental Results

We evaluate ReScaler against various poisoning and backdoor attacks in Table 1. Overall, for all attack strategies, ReScaler significantly reduces the initial attack success rate, and sends the vast majority of target images back to their ground truth classes. Meanwhile, the defense does not significantly impact the validation accuracy.

Specifically speaking, data poisoning methods (e.g., FC and BP attacks) show a strong attack capacity with a relatively high attack success rate without the defense. For example, the number of successfully attacked models is 93 among 100 models for ResNet-34 using the FC attack. When applied with ReScaler, the (average) attack success rate decreases significantly to 18.27% (from 100%), as defined in Section 4.1. ReScaler also achieves an overall high put-back rate of above 96%. With such strong defense capacity, we also show that ReScaler does not cause a large decrease on the clean validation accuracy. The results demonstrate that downweighting residual blocks using ReScaler does not hurt the model capacity to learn a mapping of clean images with their clean labels, but still effectively prevents the model from learning the poisoning mapping. The results further confirm our exploration on residual modules of poisoning and backdoor attacks.

Moreover, for clean-label backdoor attack methods, i.e., CLBD and HTBD, they already have a lower attack success rate without any defense. For example, for ResNet-34 with HTBD attack, the attack success rate without defense is only 10%, i.e., only 10 models are successfully attacked among 100 models trained with poisoning samples. Then with ReScaler, all of them are defended successfully, i.e., the defended attack success rate becomes 0%, while the de-

Attack	Model	SA	Def. ASR	Put-back Rate	Target Conf	Poison Conf	Undef. Val Acc	Def. Val Acc
FC	ResNet-34	93	18.27 ± 4.00	96.68 ± 1.30	73.90 ± 3.42	24.42 ± 3.23	93.27 ± 0.08	89.55 ± 0.26
	ResNet-50	68	14.71 ± 4.29	98.27 ± 1.71	74.26 ± 3.29	22.94 ± 2.72	93.09 ± 0.06	90.26 ± 0.13
	ResNeXt-29	82	17.07 ± 4.15	100.0 ± 0.00	70.63 ± 2.89	27.24 ± 2.46	91.19 ± 0.12	86.08 ± 0.15
BP	ResNet-34	78	29.48 ± 5.16	96.36 ± 2.52	63.67 ± 4.28	33.76 ± 4.15	92.7 ± 0.11	88.12 ± 0.30
	ResNet-50	31	29.03 ± 8.15	95.45 ± 4.44	60.34 ± 6.08	33.49 ± 5.27	93.10 ± 0.04	90.01 ± 0.13
	ResNeXt-29	41	43.90 ± 7.75	100.0 ± 0.00	47.48 ± 5.24	51.72 ± 5.20	91.21 ± 0.04	85.77 ± 0.11
CP	ResNet-34	13	15.38 ± 10.0	81.81 ± 11.6	65.98 ± 10.61	12.22 ± 6.71	93.35 ± 0.05	90.39 ± 1.46
CLBD	ResNet-34	4	25.00±21.65	66.67±27.22	51.36±21.99	28.39±14.97	92.80±0.12	86.80±0.27
	ResNet-50	2	0.00±0.00	100.00±0.00	58.26±9.12	12.13±8.23	92.84±0.06	88.85±0.21
	ResNeXt-29	6	0.00±0.00	100.00±0.00	84.93±9.88	13.07±8.03	89.15±0.16	83.73±0.13
HTBD	ResNet-34	10	0.00±0.00	66.67±19.25	43.31±13.48	23.39±8.61	93.50±0.07	89.47±0.23
	ResNet-50	2	0.00±0.00	100.00±0.00	41.31±11.31	10.23±10.49	92.51±0.09	87.74±0.27
	ResNeXt-29	11	0.00±0.00	100.00±0.00	82.27±10.99	17.70±10.99	91.42±0.04	83.23±0.21

Table 1. Results of ReScaler applied to the Poison Frog (FC), BP, CP, CLBD and HTBD attacks. 'Attack' and 'Model' indicate the attack methods and the model architectures respectively. 'SA' indicates the number of successful attacks among 100 poisoned models. 'Def. ASR', 'Put-back Rate', 'Target Conf' and 'Poison Conf' indicate the defended attack success rate, put back rate, target confidence and poison confidence. 'Undef. Val Acc' and 'Def. Val Acc' indicate the validation accuracies before or after ReScaler defense.

fended validation accuracies do not decrease much. These results further demonstrate that our proposed ReScaler can easily generalize to different attack strategies, including both poisoning and backdoor attacks.

Sensitivity of Epsilon (ϵ) Values. In Figure 2, we explore how ϵ affects the trade-off between the attack success rate and validation error for defended models (against both the FC and BP attacks). Validation error is measured using the benign test set. We measure the defense effectiveness by analyzing the trade-off between attack success rate and validation accuracy, which is controlled by the ϵ parameter. In general, by increasing epsilon, we can get attack success rates on the order of 15-30% by incurring a 3% increase in clean validation error. Further, we observe that this trade-off seems to be more favorable for deeper ResNets such as ResNet-50 (when compared to ResNet-34 and ResNeXt-29) on attacks such as FC and BP. This is evidenced by our experiments in Table 1 which show ResNet-50 achieving the lowest ASR and highest validation accuracy combination in both the FC and BP cases. As a result, we believe a practitioner can adjust ϵ and the choice of architecture to achieve a desired level of validation accuracy with our defense.

Comparison with Other Defenses. We evaluated STRIP [7], Activation Clustering (AC) [4], and Spectral Signatures (SS) [23], which are SOTA defenses. For STRIP, it fails to detect any target images during test-time for FC, CP and BP. It is because STRIP was designed for backdoor attacks with a trigger, thus it does not support the clean-label poisoning attacks in our evaluation. For AC and SS, they require access to the full training set in order to flag potential poisoning samples, remove them and re-train the model on the remaining samples. On the other hand, ReScaler does not rely on the knowledge of the training set.

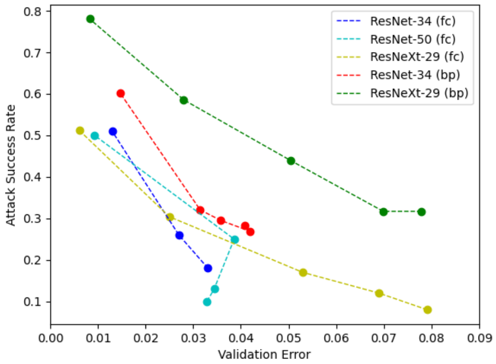

Figure 2. Effect of varying epsilon on attack success rate and validation error for FC and BP attacks. We can observe a trade-off in the figure: when ϵ increases, the ASR will decrease but the validation error will increase. More details are in the appendix.

5. Conclusion

Poisoning and backdoor attacks pose a great threat along with the enormous data requirement for training DNN models. Meanwhile, larger DNNs tend to require expensive training costs and adopt some special modules to improve gradient propagation, amongst which residual modules are the most common. In this paper, we propose ReScaler as a defense strategy by adjusting the gradient flow in residual modules via a simple test-time adaptation. Although it increases the inference time, there is no extra training costs, suggesting that ReScaler is still much faster than those training-based defense methods.

References

[1] Hojjat Aghakhani, Dongyu Meng, Yu-Xiang Wang, Christopher Kruegel, and Giovanni Vigna. Bullseye polytope: A scalable clean-label poisoning attack with improved transferability. *arXiv preprint arXiv:2005.00191*, 2020. 1, 2

[2] Thomas Bachlechner, Bodhisattwa Prasad Majumder, Huanru Henry Mao, Garrison W Cottrell, and Julian McAuley. Rezero is all you need: Fast convergence at large depth. *arXiv preprint arXiv:2003.04887*, 2020. 7

[3] Mauro Barni, Kassem Kallas, and Benedetta Tondi. A new backdoor attack in cnns by training set corruption without label poisoning. In *ICIP*, 2019. 7

[4] Bryant Chen, Wilka Carvalho, Nathalie Baracaldo, Heiko Ludwig, Benjamin Edwards, Taesung Lee, Ian Molloy, and Biplav Srivastava. Detecting backdoor attacks on deep neural networks by activation clustering. In *AAAI Workshop*, 2019. 1, 4, 7

[5] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017. 7

[6] Xinyun Chen, Wenxiao Wang, Chris Bender, Yiming Ding, Ruoxi Jia, Bo Li, and Dawn Song. Refit: a unified watermark removal framework for deep learning systems with limited data. In *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*, pages 321–335, 2021. 1

[7] Yansong Gao, Change Xu, Derui Wang, Shiping Chen, Damith C Ranasinghe, and Surya Nepal. Strip: A defence against trojan attacks on deep neural networks. In *Proceedings of the 35th Annual Computer Security Applications Conference*, pages 113–125, 2019. 4

[8] Jonas Geiping, Liam Fowl, Gowthami Somepalli, Micah Goldblum, Michael Moeller, and Tom Goldstein. What doesn't kill you makes you robust (er): Adversarial training against poisons and backdoors. *arXiv preprint arXiv:2102.13624*, 2021. 1, 7

[9] Jonas Geiping, Liam H Fowl, W Ronny Huang, Wojciech Czaja, Gavin Taylor, Michael Moeller, and Tom Goldstein. Witches' brew: Industrial scale data poisoning via gradient matching. In *ICLR*, 2020. 1, 2, 7

[10] Klaus Greff, Rupesh K Srivastava, and Jürgen Schmidhuber. Highway and residual networks learn unrolled iterative estimation. *arXiv preprint arXiv:1612.07771*, 2016. 1

[11] Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 2019. 7

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *ECCV*, 2016. 2

[13] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, 2017. 2

[14] W Ronny Huang, Jonas Geiping, Liam Fowl, Gavin Taylor, and Tom Goldstein. Metapoison: Practical general-purpose clean-label data poisoning. *Advances in Neural Information Processing Systems*, 2020. 2, 7

[15] Bargav Jayaraman and David Evans. Evaluating differentially private machine learning in practice. In *28th {USENIX} Security Symposium ({USENIX} Security 19)*, pages 1895–1912, 2019. 1

[16] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. *Technical Report, University of Toronto*, 2009. 3

[17] Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International Symposium on Research in Attacks, Intrusions, and Defenses*, 2018. 1, 2, 7

[18] Xuankai Liu, Fengting Li, Bihan Wen, and Qi Li. Removing backdoor-based watermarks in neural networks with limited data. In *ICPR*. IEEE, 2021. 7

[19] Yunfei Liu, Xingjun Ma, James Bailey, and Feng Lu. Reflection backdoor: A natural backdoor attack on deep neural networks. In *ECCV*, 2020. 7

[20] Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. Hidden trigger backdoor attacks. In *AAAI*, 2020. 1, 2, 7

[21] Avi Schwarzschild, Micah Goldblum, Arjun Gupta, John P Dickerson, and Tom Goldstein. Just how toxic is data poisoning? a unified benchmark for backdoor and data poisoning attacks. In *ICML*, 2021. 3

[22] Ali Shafahi, W Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. Poison frogs! targeted clean-label poisoning attacks on neural networks. In *NeurIPS*, 2018. 1, 2

[23] Brandon Tran, Jerry Li, and Aleksander Madry. Spectral signatures in backdoor attacks. In *NeurIPS*, 2018. 1, 4, 7

[24] Alexander Turner, Dimitris Tsipras, and Aleksander Madry. Clean-label backdoor attacks. 2018. 2, 7

[25] Andreas Veit, Michael J Wilber, and Serge Belongie. Residual networks behave like ensembles of relatively shallow networks. *Advances in neural information processing systems*, 2016. 1

[26] Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y Zhao. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *Proceedings of the IEEE Symposium on Security and Privacy*, 2019. 1, 7

[27] Dongxian Wu, Yisen Wang, Shu-Tao Xia, James Bailey, and Xingjun Ma. Skip connections matter: On the transferability of adversarial examples generated with resnets. In *ICLR*, 2020. 8

[28] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017. 2

[29] Xiaojun Xu, Qi Wang, Huichen Li, Nikita Borisov, Carl A Gunter, and Bo Li. Detecting ai trojans using meta neural analysis. In *Proceedings of the IEEE Symposium on Security and Privacy*, 2021. 1

[30] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016. 7

[31] Chiyuan Zhang, Samy Bengio, and Yoram Singer. Are all layers created equal? *arXiv preprint arXiv:1902.01996*, 2019. 1, 2

[32] Pu Zhao, Pin-Yu Chen, Payel Das, Karthikeyan Natesan Ramamurthy, and Xue Lin. Bridging mode connectivity in loss landscapes and adversarial robustness. In *ICLR*, 2019. 7

[33] Chen Zhu, W Ronny Huang, Hengduo Li, Gavin Taylor, Christoph Studer, and Tom Goldstein. Transferable clean-label poisoning attacks on deep neural nets. In *International Conference on Machine Learning*, pages 7614–7623. PMLR, 2019. 2