Dartmouth College

Dartmouth Digital Commons

Dartmouth College Undergraduate Theses

Theses and Dissertations

6-1-2020

Query Free Adversarial Transfer via Undertrained Surrogates

Christopher S. Miller
Dartmouth College

QUERY FREE ADVERSARIAL TRANSFER VIA UNDERTRAINED SURROGATES

Senior Honors Thesis

Dartmouth Computer Science Technical Report TR2020-889

Author

Christopher Miller

Advisor

Soroush Vosoughi

DARTMOUTH COLLEGE

Hanover, New Hampshire

June 2020

Abstract

Adversarial examples consist of minor perturbations added to a model's input which cause the model to output an incorrect prediction. Deep neural networks have been shown to be highly vulnerable to these attacks, and this vulnerability represents both a security risk for the use of deep learning models in security-conscious fields and an opportunity to improve our understanding of how neural networks generalize to unexpected inputs. Transfer attacks are an important subcategory of adversarial attacks. In a transfer attack, the adversary builds an adversarial attack using a surrogate model, then uses that attack to fool an unseen target model. Recent research in this subfield has focused on attack generation methods which can improve transferability between models and ensemble-based attacks. We show that optimizing a single surrogate model is a more effective method of improving adversarial transfer, using the simple example of an undertrained surrogate. This method of attack generation transfers well across varied architectures and outperforms state-of-the-art methods. To interpret the effectiveness of undertrained surrogate models, we represent adversarial transferability as a function of surrogate model loss function curvature and surrogate gradient similarity to target gradient and show that our approach reduces the presence of local loss maxima which hinder transferability. Our results suggest that finding good single surrogate models is a highly effective and simple method for generating transferable adversarial attacks, and that this method represents a valuable route for future study in this field.

Contents

Chapter 1

Introduction

Previous work has shown that deep learning models are vulnerable to adversarial perturbations [26]. These are small modifications to an input image which cause the model to output an incorrect prediction. Adversarial attacks take many forms. The primary divide is between white box and black-box attacks. In white box attacks, the attacker has access to the parameters and architecture of the target model, allowing them to utilize model gradients and losses. In black-box attacks, the adversary has no access to the parameters of the target model, and may or may not have access to its architecture or use of the target as an oracle model (which allows the adversary to submit inputs and receive model predictions on those inputs).

Understanding adversarial examples is an important task. When deep models are applied in security-conscious domains such as autonomous driving, healthcare, and fraud detection, their vulnerabilities to attack become vulnerabilities which can threaten individual health and safety. This takes on special importance with the advent of work showing that adversarial examples can be engineered to fool models in the physical world [15, 6]. Understanding adversarial examples may suggest methods for producing defenses against them. Currently the most consistently successful method for defending models is adversarial training, which entails training on adversarial examples in addition to (or instead of) clean images [7]. Robust adversarial training methods are computationally expensive, reduce model accuracy on clean inputs, and only provide limited security, suggesting that better methods are needed [7, 19]. Adversarial examples can also be used to fool malicious neural networks. [23] shows that adversarial examples can cause generative adversarial networks to output unrealistic images, a potential defense against deepfake-style malicious image translation.

Section 1.1
Adversarial transfer

Szegedy et al. showed that adversarial examples also have the ability to transfer between models, allowing an example generated on one model to fool a different model [26]. This effect means that keeping a model's parameters and query access private is not an effective way to protect a model from adversarial attack. As long as an adversary has the ability to train a model on a compatible dataset, the adversary can use their own model as a surrogate to produce adversarial examples which may then fool the target classifier.

Many methods for producing effective black-box transfer rely on access to the outputs of the target model [1]. Access to target model outputs allows for a class of attacks known as gradient estimation attacks, in which the adversary attempts to estimate the gradients of the target model in order to approximate a white-box attack in a black-box setting. Required access to the target model can range from the predicted label for an input to class-conditional probability predictions for all classes. Recent approaches to gradient-estimation attacks include [3],[13], [27], and [2]. Other approaches such as [20] use query access to construct adversarial examples without gradient estimation.

This approach is not always representative of real-world model access. In real-world settings, models may be secured by restricting access to outputs or limiting the permitted number of queries. In these settings, an adversary may not have access to the predictions for any inputs. This led Ilyas et al. to introduce more restrictive black-box settings in [12].

We use the restrictive zero-knowledge, zero-access setting, in which an adversary has no knowledge of model architecture or parameters, and has no access to model outputs. An adversary which can effectively fool a target model in this setting is extremely strong.

One approach in this setting focuses on building attacks on ensembles of surrogate classifiers [18]. This approach reduces the overfit of attacks to the surrogate model, and increases their ability to transfer to the target model. A limitation of this approach is the computational requirements for an adversary to train multiple diverse classifiers and build adversarial examples using them.

Recent attempts to produce more successful transfer attacks have focused on creating stronger attack generation methods. The current state of the art in query-free adversarial transfer is the Intermediate Level Attack (ILA) introduced by Huang et al. in [11]. They show that enhancing a previously generated adversarial example by increasing its perturbation on a certain layer of the surrogate model substantially improves transfer to target models.

Our work, by contrast, focuses on finding a more effective individual surrogate model. We show that generating simple attacks on a more effective surrogate produces stronger transferability than generating more sophisticated attacks on a less effective surrogate. We suggest that further research into finding highly effective surrogate models may be a promising avenue for producing strong transfer attacks and accurately assessing the true robustness of existing models to black-box attack.

Chapter 2

Methodology

Section 2.1

Data

We use the CIFAR-10 dataset for training and testing [14]. This dataset is composed of 60,000 images divided into ten disjoint classes. There are 6,000 images per class, and the dataset is divided into 50,000 training images and 10,000 test images. Each image is a full color $3 \times 32 \times 32$ image.

Section 2.2

Models

We chose to evaluate our approaches across a wide variety of model architectures to produce an accurate assessment of transfer between both similar and different models. We report results using ResNet18 models, SENet18 models, GoogLeNet models, DenseNet121 models, and MobileNetV2 models as described by [8, 9, 25, 10, 24]. Model architectures are as they is defined in their original papers, with minor changes given by [17] to accommodate the input size of CIFAR-10 images. All models are trained and evaluated on an individual NVIDIA Tesla P100 GPU.

2.2.1. Natural training

For naturally trained models, we used the definitions and training scripts provided by [17] and trained the models for 90 epochs. We used a stepped learning rate schedule with an initial learning rate of 0.1, decreased by a factor of 10 at epochs 30 and 60. We achieve final accuracy of 94.29% for the ResNet18 model, 94.44% for the SENet18 model, 94.43% for the GoogLeNet model, 92.23% for the MobileNetV2 model, and 94.99% for the DenseNet121 model. Our trained accuracies are comparable to those reported by [17].

To ensure that our analysis is comparable to the current state of the art, we use the pretrained target models released by [11]. These models are a ResNet18 model, an SENet18 model, a DenseNet121 model, and a GoogLeNet model as described by [8, 9, 10, 25]. As with our surrogate models, these models are as defined by their original authors with the exception of minor modifications to support the 32×32 input size of CIFAR-10 images. These models are implemented and trained using code released by [17], consistent with our training. The authors train their models for 350 total epochs, using a learning rate schedule which starts at 0.1 and reduces by a factor of 10 at epochs 150 and 250. The ResNet model is trained to 94.76% accuracy on the CIFAR-10 validation set, the SENet model is trained to 94.58% accuracy, and the GoogLeNet model is trained to 94.85% accuracy. To introduce a more distinct model architecture, we also train a MobileNetV2 model for 350 epochs using the same learning rate schedule. This model achieves final test set accuracy of 94.06%.

2.2.2. Adversarial training

For adversarially trained models, we use fast adversarial training, introduced by [29], with $\epsilon = .05$ to train ResNet18 models via adversarial training. Fast adversarial training introduces FGSM adversarial examples with random initializations into each minibatch while training, controlling for a phenomenon the authors refer to as catastrophic overfitting to produce robustness to iterative attack. Following the parameters introduced by [29], we train robust models for 45 epochs using cyclic learning rates. Our first model achieves clean test accuracy of 79.66%, and our second achieves test accuracy of 79.58%. The first model achieves white box adversarial accuracy of 41.03% against a 20-iteration I-FGSM attack, and the second gets 40.98% accuracy. These results are comparable to those reported by [29], and show that the adversarial training method makes these models quite robust to adversarial examples.

Section 2.3

Attacks

We evaluate our models against a variety of attacks, and show them below. Our results are based on standard implementations of the attacks released by Huang et al. and the Cleverhans team [11, 22]. Unless otherwise indicated, $\epsilon = .05$ for all attacks.

The **Fast Gradient Sign Method** (FGSM) was introduced by Goodfellow et al. as a simple method for producing adversarial examples efficiently [7]. The adversarial perturbation is generated by scaling the sign of the model's gradient by ϵ, and this perturbation is added to the original image to form the adversarial example. The adversarial image, $\hat{x}$, is defined by 2.1, where $\nabla\ell(x)$ is the gradient of the model loss with respect to input x.

$$\hat{x} = x + \epsilon \cdot \mathrm{sign}(\nabla\ell(x)) \tag{2.1}$$

Iterative FGSM (I-FGSM), also referred to as the Basic Iterative Method, is a simple extension to the FGSM attack, introduced by [15]. This method applies

FGSM repeatedly to produce a more finely targeted adversarial example. The attack is defined in Equation 2.2, where $\hat{x_n}$ indicates the adversarial example produced by n steps of I-FGSM and α indicates the learning rate. Here the function Clip restricts the adversarial example to remain within the ϵ-ball surrounding x.

$$\hat{x_n} = \text{Clip}_{x,\epsilon}(\hat{x_{n-1}} + \alpha \cdot \text{sign}(\nabla \ell(x_{n-1}))) \tag{2.2}$$

We use a learning rate (the epsilon value during each iteration) of .005 and 20 iterations. These values were determined empirically via grid search to produce the strongest transfer.

Momentum I-FGSM (MI-FGSM) was introduced by [5] to enhance iterative attack transferability. The authors find that incorporating a momentum term in when calculating I-FGSM increases the stability of the attack by reducing its susceptibility to being trapped in a local loss maximum. We use a learning rate of .005 and 20 iterations of attack, with a decay $\mu = 0.9$.

The **Transferable Adversarial Perturbation** (TAP) attack, introduced by Zhou et al., uses intermediate feature representations to generate an adversarial example [32]. The TAP attack attempts to maximize the distance between the original image and the adversarial image in the intermediate feature maps of the surrogate model. The authors also show that applying smooth regularization to the resulting perturbation improves transfer between models.

The **Intermediate Level Attack** (ILA) was introduced by Huang et al. in 2019 as the state of the art in query-free transfer attacks [11]. This method takes a predefined adversarial example, created using another method, and enhances its perturbation on intermediate layer representations in the surrogate model. The attack uses the predefined example as a guide towards an adversarial direction in which to enhance the intermediate layer disturbance.

We refer to an ILA attack which enhances an example produced by FGSM as ILA-enhanced FGSM, and follow the same convention for other ILA attacks. For ILA-enhanced iterative attacks, we follow the methodology of the original paper and use ten iterations of the original attack followed by ten iterations of ILA enhancement.

For each surrogate, we evaluated ILA attacks based on each possible layer, and found that the optimal source layer to enhance perturbations on was consistent for all epochs of the surrogate model. We find that the optimal layer to target is block 4 for ResNet18 and SENet18, block 0 for MobileNetV2, block 9 for GoogLeNet, and block 6 for DenseNet121, and we report results for ILA based on these layers. These parameters are consistent with the optimal target layers shown by [11].

Chapter 3

Approach

Our approach diverges from previous work focused on developing stronger and more transferable methods for generating adversarial attacks. These include algorithmic approaches such as MI-FGSM, which introduced a momentum term to prevent the attack from being caught in local loss maxima, and vr-IGSM, which introduced local gradient smoothing for the same reason [5, 30]. They also include novel approaches such as input diversity, in which the attacker transforms the adversarial image during the attack to find a more generalizable perturbation [31].

We take a reversed approach, and attempt to develop a surrogate model with a generalizable yet low complexity decision boundary. The surrogate model must be generalizable to approximate the decision boundary of the data manifold well (and thus approximate the decision boundary of the target model well). It must also be low complexity, to limit the effect of local loss maxima on the generated attack.

To maximize generalization of the surrogate model while minimizing decision boundary complexity, we aim to minimize overfit of the surrogate model on the training data. If we avoid the surrogate model overfitting to the training data, we can in turn ensure that it learns a highly generalizable, low-complexity function. Learning this type of function maximizes our chance of an attack on the surrogate model generalizing well to the target model, as shown by [4].

These observations give rise to a simple method for minimizing overfit to the training data: undertraining the surrogate model. We define an undertrained surrogate model as the condition of the model at an earlier epoch with a lower validation set accuracy than the final, fully trained model (ie, the model at the point which has the highest validation set accuracy). Note that we require both conditions, as an epoch with lower validation set accuracy that comes after the epoch with the best accuracy is likely to be overtrained rather than undertrained, and by definition of the fully trained epoch no other epoch can have a higher validation set accuracy.

To evaluate this approach, we train surrogate models fully as described in Section 2.2 and save a copy of the model parameters after each epoch of training. This enables retroactive evaluation of the optimal point for building an adversarial attack. We

We show an overview of our transfer evaluation methodology in Figure 3.1.

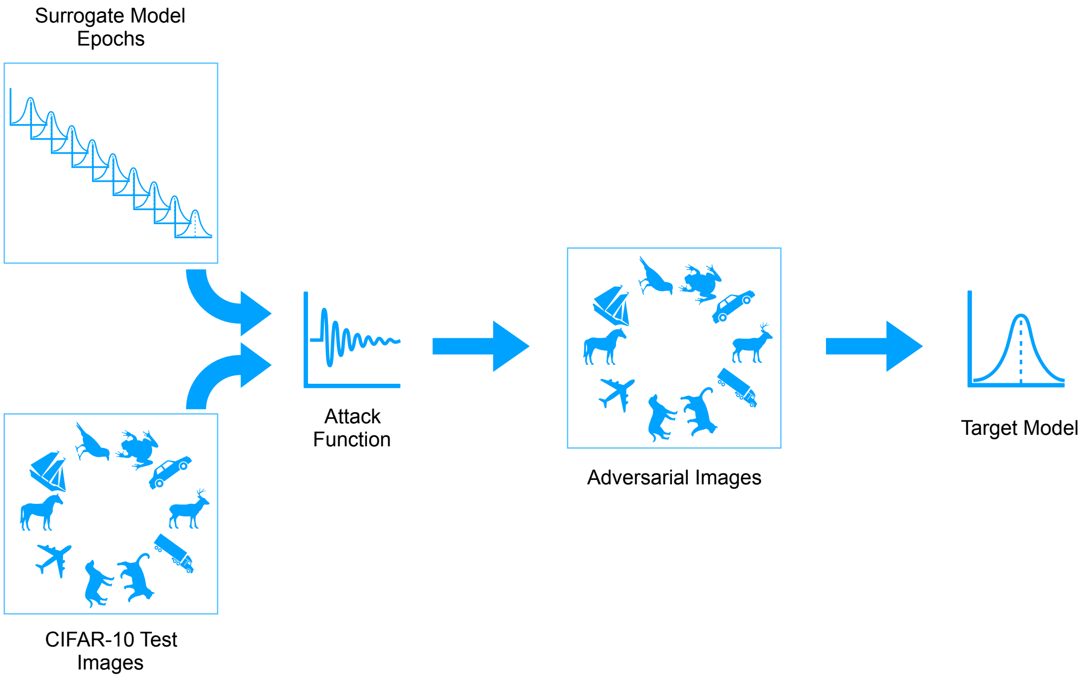


Figure 3.1: Overview of our methodology for evaluating undertrained adversarial transfer to a target model.

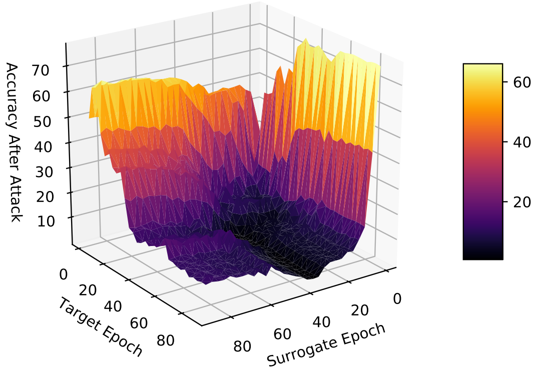


Figure 3.2: ResNet18 post-attack accuracy on I-FGSM.

Chapter 4

Results

Section 4.1

Natural training transfer

To evaluate the effectiveness of this approach, we tested transfer between two separately trained ResNet18 models across epochs. Both models were trained using the surrogate model training procedure described in 2.2. We evaluated transfer by generating attacks on every third epoch of the first model (ie, the epoch set {1, 4, 7, ..., 88}) and transferred them to the all epochs in the same set of the second model. Our results are shown in Figure 3.2, where lower accuracy after attack indicates more effective transfer. Although intuition would suggest that attacks generated on a given surrogate epoch would transfer best to the same epoch of target model, our results show that this is not the case. The resulting accuracy landscape instead shows a distinct valley between surrogate epochs 20 and 40, where the same surrogate models transfer well to almost all target model epochs. Also significant is that these epochs outperform later epochs (those which are more fully trained) in transferring to target models. These results validate the hypothesis that undertraining can produce a surrogate model which generates significantly more transferable adversarial examples.

We expand our analysis by evaluating transfer from intermediate epochs of ResNet18, SENet18, MobileNetV2, GoogLeNet, and DenseNet121 models to separately trained target models from the same set of architectures. We consider this wide variety of architectures to evaluate the effectiveness of our approach in a true black-box setting, where the target model architecture is unknown and thus no guarantees can be made about the similarity of the target and surrogate model architectures. Note that the separate training means that attacks between the same architectures are not strictly white box attacks, as the models have different parameters. We generate attacks on every other epoch of the surrogate model, and target the epoch of the target model with the lowest test loss (ie, the fully trained final model). We evaluate transfer for a variety of attacks: FGSM, I-FGSM, MI-FGSM, ILA-enhanced FGSM, ILA-enhanced I-FGSM, and Transferable Adversarial Perturbations. For all attacks, we use the parameters outlined in Section 2.3, and use the attack method to form adversarial examples based on all 10,000 images in the CIFAR-10 validation set.

We compare our approach of generating adversarial examples on intermediate epochs to the previous standard approach of generating adversarial examples on the best loss model. Our findings suggest that the intermediate-epoch approach produces examples which transfer substantially more successfully than previous approaches across a variety of attack styles. For all attacks evaluated, the intermediate epoch attack outperforms the best epoch attack.

Our results for the best intermediate epoch compared to the previous approach using a ResNet18 surrogate model are given in Table 4.2, and we show results for all surrogate models and all attacks in Figures 4.1-4.4. We provide the equivalent tables for other surrogates and the equivalent graphs for other attacks ein the supplementary material.

We show that an intermediate epoch MI-FGSM attack produces the strongest results across most surrogate-target combinations, with the exception of the ResNet18 surrogate. We discuss potential reasons for I-FGSM performing similarly to MI-FGSM on the ResNet18 surrogate in 5.

We selected the strongest surrogate (ResNet18) by choosing the surrogate which produced attacks which performed the best on average across all target models. We choose to focus on the MI-FGSM attack, since it performs the best across the full range of surrogate models. MI-FGSM attacks based on the ResNet18 surrogate reduced accuracy on ResNet18 by 97.65%, on GoogLeNet by 94.07%, on MobileNetV2 by 97.02%, on SENet18 by 97.64%, and on DenseNet121 by 97.66%.

These results emphasize the exceptional ability of intermediate attacks to generalize across a wide variety of target model architectures in a black-box setting. To the best of our knowledge, this makes an intermediate epoch MI-FGSM attack (IMI-FGSM) the state of the art in query-free transfer attacks.

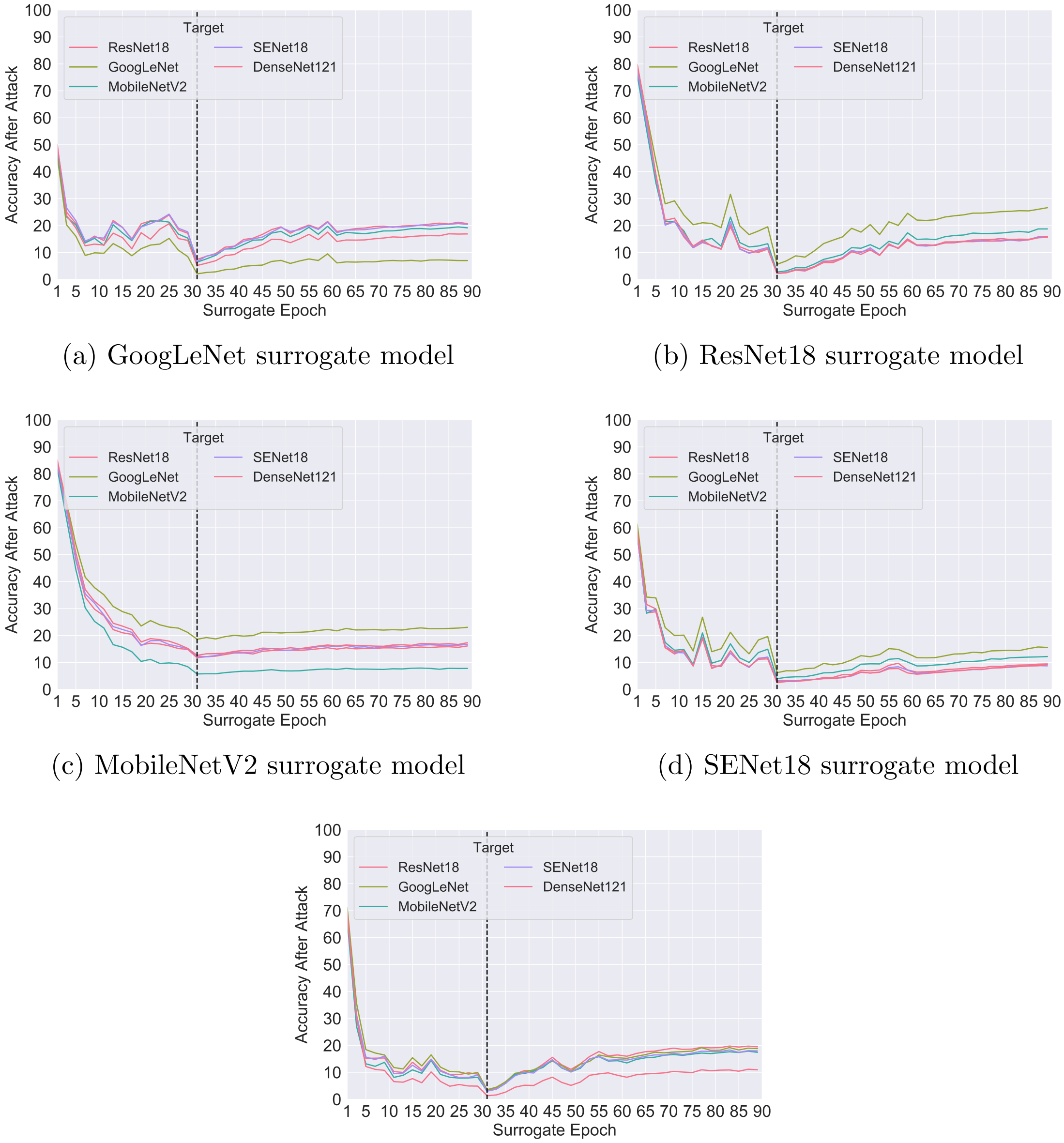
Our results also show that the optimal surrogate epoch for transfer is consistent across target models. This indicates that an adversary can select the strongest surrogate by evaluating performance on other models, without requiring any query access to the target model, fitting our case of the black-box setting with zero query access.

To address the possibility that our final epoch models are overtrained, and do not represent typical models, we also perform attacks between our target models (i.e., we use each target model as a surrogate, and evaluate the effectiveness of attacks generated on the target model when transfering to other target models). All models except for the MobileNetV2 architecture were released as pretrained models by [11], who used these models as their target and surrogate models. All including the MobileNetV2 architecture were trained using a open-sourced training code and hyperparameters provided by [17]. When evaluated as surrogates, these models perform very similarly to the best loss epochs of our separately trained surrogate models, and our undertrained models outperform them. This indicates that our best loss models are representative of typical models trained on the CIFAR-10 dataset. We report full results of this analysis in Table 4.1.

Table 4.1: Accuracy (%) after attack by given target model (column) for attacks generated using the given source model. We omit the diagonal, since the diagonal is a white box attack. $\epsilon = .05$. Lower accuracy indicates better transfer.

Attack	Source	Target				
		ResNet18	SENet	MobileNet	GoogLeNet	DenseNet
MI-FGSM	ResNet18	—	17.90	21.87	29.06	19.64
	SENet18	12.28	—	16.37	20.10	12.87
	MobileNetV2	17.64	18.31	—	21.89	16.78
	GoogLeNet	26.80	18.31	27.87	—	22.93
	DenseNet121	11.42	10.78	12.13	12.67	—
I-FGSM	ResNet18	—	27.35	31.45	42.81	29.34
	SENet18	16.06	—	21.00	25.88	17.26
	MobileNetV2	23.41	24.60	—	28.33	22.07
	GoogLeNet	36.90	35.61	36.73	—	31.29
	DenseNet121	21.65	20.72	21.95	24.10	—
ILA	ResNet18	—	7.95	9.60	12.96	8.41
MI-FGSM	SENet18	7.07	—	8.56	11.29	7.36
	MobileNetV2	27.24	29.00	—	33.45	26.60
	GoogLeNet	11.47	11.56	12.19	—	8.56
	DenseNet121	6.91	6.59	7.01	7.87	—
ILA	ResNet18	—	7.67	9.35	13.86	7.74
I-FGSM	SENet18	6.66	—	8.14	10.98	6.88

	MobileNetV2	28.04	29.66	—	34.17	26.94
	GoogLeNet	12.52	12.14	13.05	—	9.05
	DenseNet121	6.43	6.16	6.73	7.62	—
TAP	ResNet18	—	20.66	21.55	25.70	20.72
	SENet18	15.14	—	18.37	21.58	15.81
	MobileNetV2	28.92	29.64	—	34.65	27.42
	GoogLeNet	17.79	17.82	17.10	—	15.37
	DenseNet121	20.37	19.49	20.15	20.46	—



(a) GoogLeNet surrogate model

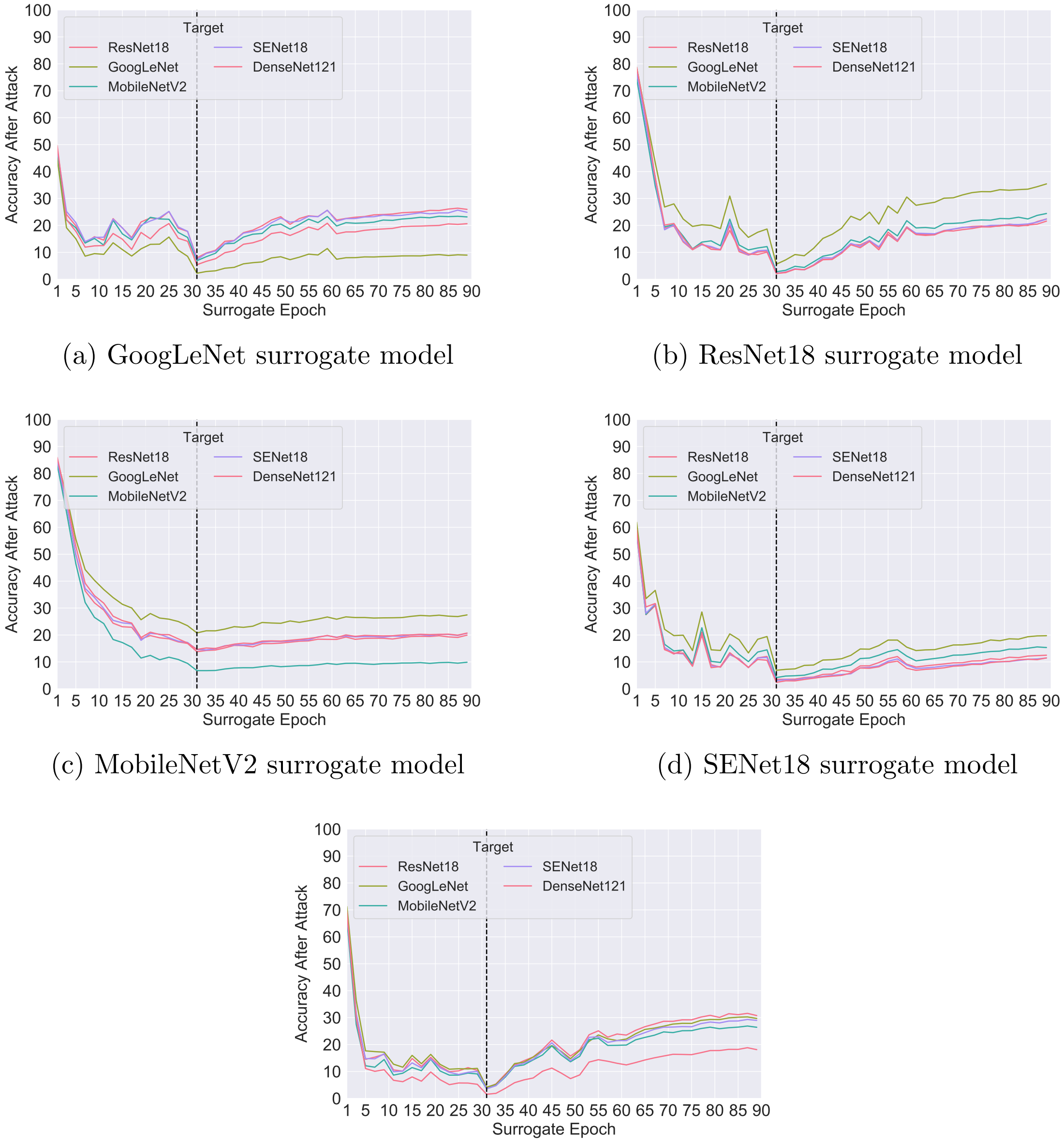
(b) ResNet18 surrogate model

(c) MobileNetV2 surrogate model

(d) SENet18 surrogate model

(e) DenseNet121 surrogate model

Figure 4.1: Post-attack accuracy for MI-FGSM transfer attacks from naturally trained models to target models, ($\epsilon = .05$). Lower accuracy indicates better transfer. The dashed lines indicate the surrogate epoch with the best transferability for most attacks.



(a) GoogLeNet surrogate model

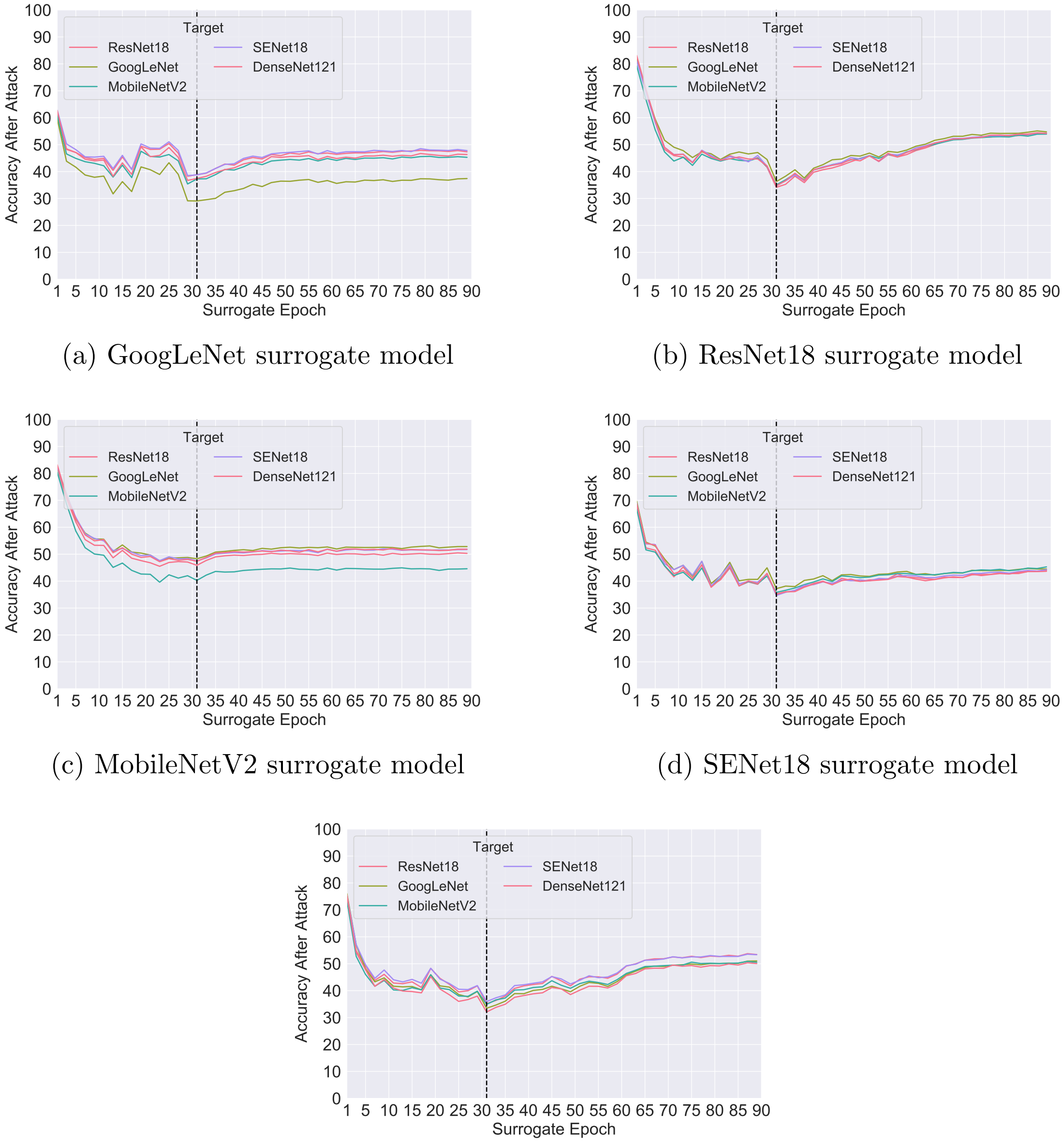
(b) ResNet18 surrogate model

(c) MobileNetV2 surrogate model

(d) SENet18 surrogate model

(e) DenseNet121 surrogate model

Figure 4.2: Post-attack accuracy for I-FGSM transfer attacks from naturally trained models to target models, ($\epsilon = .05$). Lower accuracy indicates better transfer. The dashed lines indicate the surrogate epoch with the best transferability for most attacks.



(a) GoogLeNet surrogate model

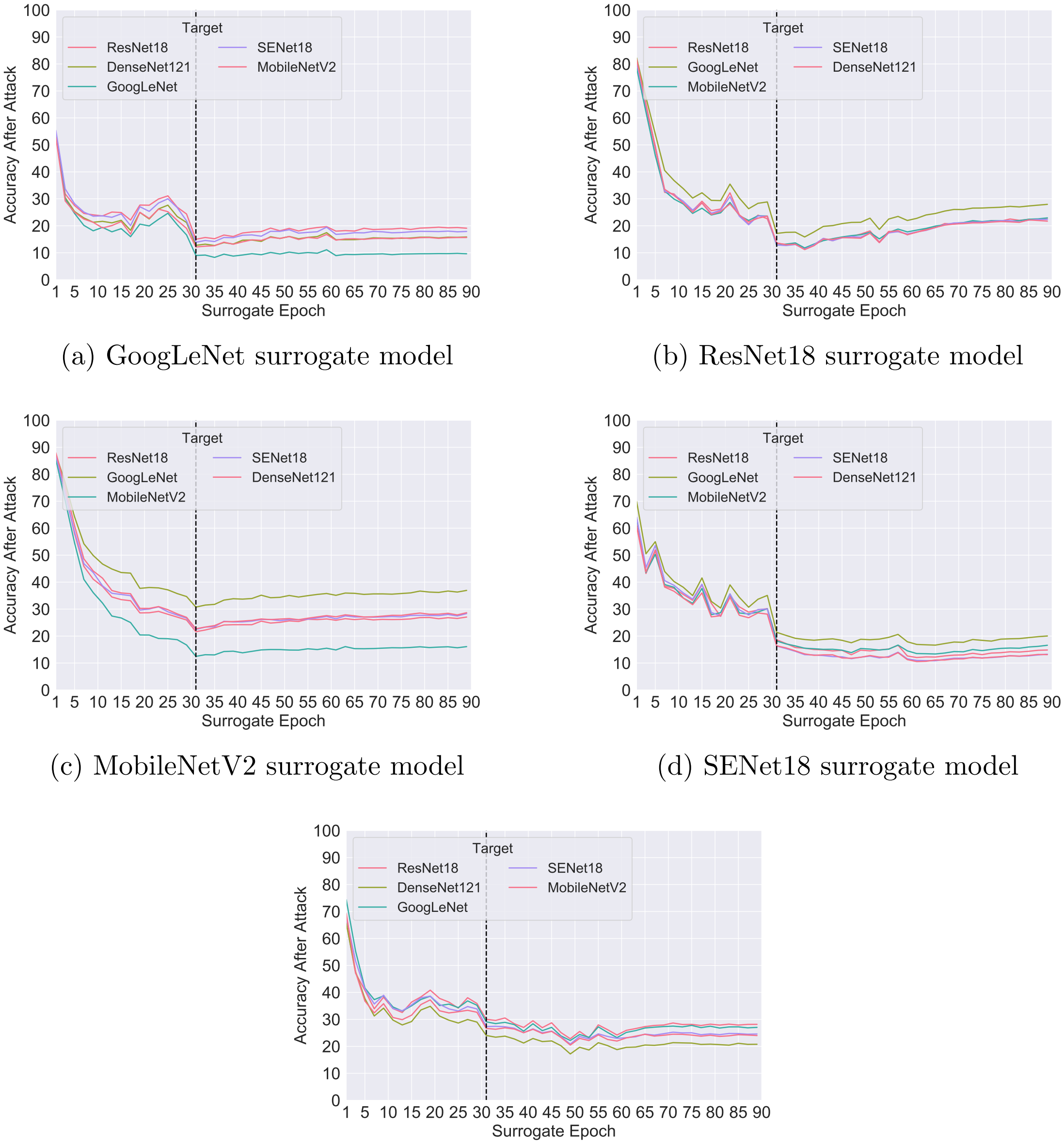
(b) ResNet18 surrogate model

(c) MobileNetV2 surrogate model

(d) SENet18 surrogate model

(e) DenseNet121 surrogate model

Figure 4.3: Post-attack accuracy for FGSM transfer attacks from naturally trained models to target models, ($\epsilon = .05$). Lower accuracy indicates better transfer. The dashed lines indicate the surrogate epoch with the best transferability for most attacks.



(a) GoogLeNet surrogate model

(b) ResNet18 surrogate model

(c) MobileNetV2 surrogate model

(d) SENet18 surrogate model

(e) DenseNet121 surrogate model

Figure 4.4: Post-attack accuracy for TAP transfer attacks from naturally trained models to target models, ($\epsilon = .05$). Lower accuracy indicates better transfer. The dashed lines indicate the surrogate epoch with the best transferability for most attacks.

As emphasis of the strength of undertrained transfer, we also compare the transfer of an intermediate attack on ResNet18 to MobileNetV2 to the transfer between two fully trained MobileNetV2 models. Our results show that the best loss I-FGSM

Table 4.2: Accuracy (%) after attack by target model for attacks generated using an intermediate surrogate epoch of a ResNet18 model (Undertrained Attack) and the surrogate epoch with the lowest validation loss (Best Loss Attack). $\epsilon = .05$. Lower accuracy indicates better transfer.

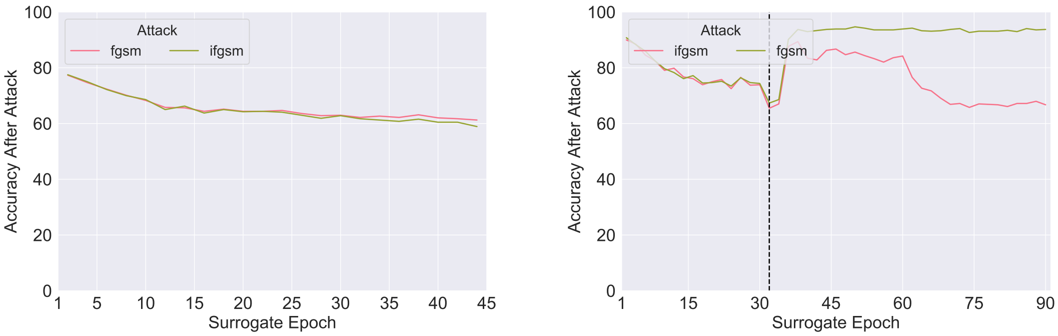
Attack	Target	Undertrained Attack	Best Loss Attack
MI-FGSM	ResNet18	**2.23**	15.63
	GoogLeNet	**5.62**	26.64
	SENet18	**2.23**	16.00
	MobileNetV2	**2.74**	18.76
	DenseNet121	**2.24**	15.87
I-FGSM	ResNet18	**2.20**	22.40
	GoogLeNet	**5.61**	35.42
	SENet18	**2.79**	24.43
	MobileNetV2	**2.79**	22.36
	DenseNet121	**2.13**	21.60
ILA MI-FGSM	ResNet18	**2.41**	10.39
	GoogLeNet	**6.35**	17.09
	SENet18	**2.69**	10.87
	MobileNetV2	**2.82**	11.99
	DenseNet121	**2.57**	10.29
ILA I-FGSM	ResNet18	**2.36**	9.98
	GoogLeNet	**6.04**	16.76
	SENet18	**2.88**	11.57
	MobileNetV2	**2.62**	10.24
	DenseNet121	**2.44**	9.42
TAP	ResNet18	**13.44**	22.83
	GoogLeNet	**17.18**	27.96
	SENet18	**12.92**	22.36
	MobileNetV2	**12.76**	22.93
	DenseNet121	**13.67**	21.74

attack generated on MobileNetV2 produces post-attack accuracy of 6.09%, and the best loss FGSM attack produces post-attack accuracy of 31.25%. Note that intermediate attacks against ResNet18, a model with substantially more parameters and one which lacks many architectural features of MobileNetV2 (depth-wise convolutions, linear bottlenecks, inverted residual blocks, etc.), transfer more successfully to the MobileNetV2 model, producing post-attack accuracies of 1.41% and 30.78% for I-FGSM and FGSM respectively.

These results show that an undertrained surrogate of completely different architecture can outperform a fully trained surrogate of the same architecture: ie, a purely black box undertrained approach outperforms a fully trained gray box approach. This suggests that undertraining a surrogate model is a highly effective strategy for producing adversarial examples which can transfer well across varied model architectures, making them ideal for an adversary in a black-box setting.

Section 4.2

Adversarially trained transfer



(a) FGSM Trained (Robust)

(b) FGSM Trained (Non-Robust)

Figure 4.5: Accuracy of an adversarially trained ResNet18 model against transferred I-FGSM examples from a second adversarially trained ResNet18 model by epoch of the surrogate model, broken out by type of adversarial training. Lower accuracy indicates better transfer.

We next investigated the impact of adversarial training on this phenomenon, providing results for transferred attacks between two ResNet18 models adversarially trained using the Fast-FGSM method introduced by [29], as discussed in Section 2.3. Some prior work has been done here by Vivek et al., who showed intermediate-epoch adversarial efficacy for non-robust adversarially trained models trained with FGSM examples (discussed there in the context of reducing the cost of ensemble adversarial training) [28]. Our results show that intermediate epoch transferability is restricted to non-robust models such as naturally trained or non-robust FGSM trained models. We report full results on adversarially trained models in Figure 4.5. Our results also call into question their conclusions that this effect is caused by adversarial training producing models which generate weak adversaries, since we find that this effect is not present in robust models.

Chapter 5

Explaining transferability

Here we evaluate potential causes of improved transferability for certain surrogates, and build an explanatory model for transferability.

Section 5.1

Gradient similarity

Gradient similarity is an important factor in transferability. Models which produce highly similar gradients on input images will produce highly similar adversarial images when using gradient-based attacks. We suggest that by undertraining the surrogate classifier, we retain more universal characteristics in the gradient, ensuring that the surrogate model gradients will be similar to any target classifier which learns the data manifold regardless of architecture and training method.

To evaluate this effect, we calculated the cosine similarity of vectorized gradients for surrogate and target models on clean images. To do so, we calculate gradients on test set images for surrogate and target models. We then reshape each image gradient into a vector, and take the l_2 normalized dot product of the target and surrogate model gradients, averaging this value across all images. Here x_s^i represents the reshaped gradient of image i with respect to surrogate model loss and x_t^i represents the same quantity with respect to target model loss:

$$Similarity = \frac{1}{n}\sum_{i=1}^{n}(\frac{x_s^i x_t^i}{|x_s^i|_2 |x_t^i|_2}) \tag{5.1}$$

We show results for all surrogate and target models in Figure 5.1. The variance in these figures shows clear correlations with the variance in transferability shown in Figure 4.1, as do more general patterns such as the relatively smooth curves for the MobileNetV2 surrogate model.

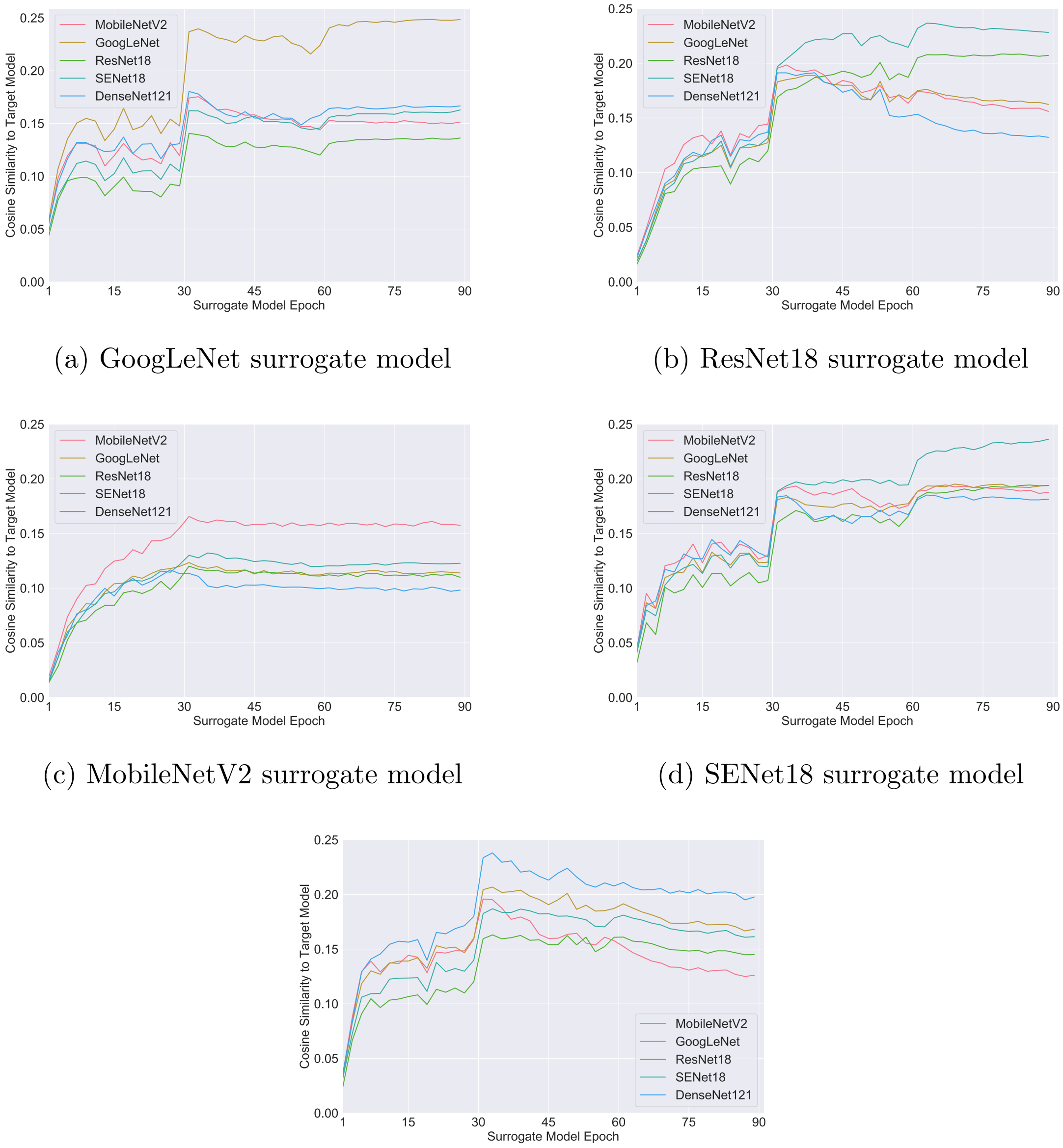
However, the results shown in Figure 5.1 raise several questions which suggest that gradient correlation is not a perfect proxy for transferability. Note that while the similarity plots for all surrogate/target models exhibit sharp spikes in gradient similarity after epoch 30, which matches transferability improving at that epoch, in many cases gradient similarity does not decline appreciably after the epochs of maximal transferability. Relying on solely gradient direction similarity, we might expect that later epochs would transfer better for some models.

These results confirm those of Liu et al. in showing that while gradient similarity between a surrogate and target model clearly has some correlation with adversarial transferability, other factors appear to be at play [18].

Section 5.2

Loss function curvature analysis

Our secondary goal in undertraining is to reduce decision boundary complexity, limiting the effect of local loss maxima on the surrogate model. While decision boundary complexity is computationally challenging to directly quantify, the topology of the decision of a network is closely related to its loss landscape, which we can measure locally [16]. Some prior work has also suggested that local loss function smooth-



(a) GoogLeNet surrogate model

(b) ResNet18 surrogate model

(c) MobileNetV2 surrogate model

(d) SENet18 surrogate model

(e) DenseNet surrogate model

Figure 5.1: Cosine similarity of gradients between each surrogate model and the target models by epoch.

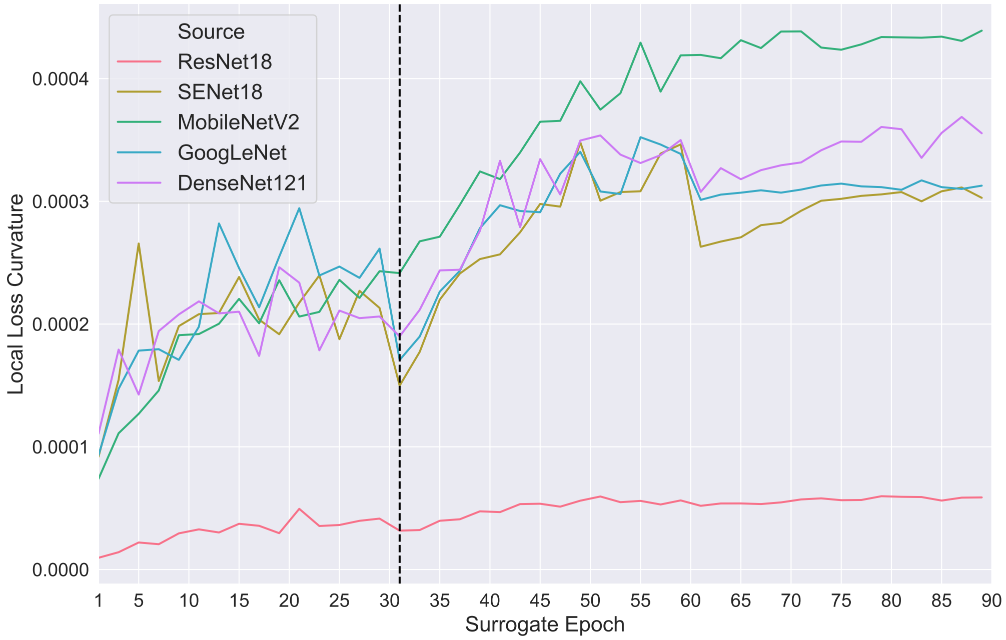


Figure 5.2: Loss curvature approximation for all surrogate models.

ness is related to adversarial transferability, and it also appears to be highly related with adversarial robustness [30, 21]. We evaluate the hypothesis that intermediate epochs have lower complexity than fully trained models. We follow [21] in using a finite difference approximation of the Hessian matrix to quantify local loss curvature. Our results show a local minimum in local loss curvature at the optimal intermediate epoch for all models except MobileNetV2. For all models considered, the optimal intermediate epoch curvature is significantly lower than the final model curvature. Figure 5.2 shows the local loss curvature by epoch for all surrogate models.

To confirm that the reduced curvature is accurately reflecting a reduction in local loss maxima, we note that the improved performance of MI-FGSM over I-FGSM is based on the ability of MI-FGSM to escape local loss maxima. We thus expect that MI-FGSM will outperform I-FGSM more on models with more local loss maxima (ie, models with a more complex loss landscape). This suggests that if higher curvature indeed implies more local loss maxima and a more complex loss landscape, it will have strong positive correlation with how much MI-FGSM outperforms I-FGSM. We calculate the Pearson correlation coefficient between curvature and how strongly MI-FGSM outperformed I-FGSM (ie, the accuracy of target models on MI-FGSM examples minus their accuracy on I-FGSM examples) for each surrogate model. We found highly statistically significant positive correlation between curvature and how strongly MI-FGSM outperformed I-FGSM (mean coefficient = 0.86, $p < .001$ for all surrogate models). This indicates that our approach is successfully producing a surrogate model with a less complex loss landscape.

These results also suggest a potential explanation for why the ResNet18 surrogate model outperforms other models in intermediate transfer; despite the relatively high parameter count, its complexity at all epochs is significantly lower than that of the other models we consider. This result may also explain why ResNet18 is the only surrogate model architecture in which an intermediate I-FGSM attack performs as well or better than the MI-FGSM attack: the lower complexity of the ResNet18 model loss landscape reduces the impact of local loss maxima, removing the advantage of the MI-FGSM.

Section 5.3
Modeling

To evaluate the impact of gradient similarity and curvature, we build an ordinary least squares linear regression model to predict post-attack accuracy. The model includes gradient similarity, squared gradient similarity, curvature, and a constant for each source model. We report coefficients and significance measures in Table 5.1. Our model produces an R2 value of 0.681, and confirms our hypothesis that gradient

Table 5.1: Linear Model

Variable	Coefficient	P-Value
Loss Curvature Magnitude	4.25	$p < .001$
Gradient Similarity	-37.10	$p < .001$
Gradient Similarity Squared	28.76	$p < .001$
DenseNet121 Source	19.99	$p < .001$
GoogLeNet Source	17.50	$p < .001$
MobileNetV2 Source	13.95	$p < .001$
ResNet18 Source	27.49	$p < .001$
SENet18 Source	15.58	$p < .001$

similarity and curvature are highly statistically significant predictors of transferability.

Chapter 6

Discussion

We show that a surrogate-focused approach to adversarial transfer outperforms attack-focused approaches We explain intermediate epoch transferability as the result of two effects: high gradient similarity and low loss function curvature. We show that gradient similarity between the surrogate and target classifier and loss function curvature are highly significant ($p < .001$) predictors of transferability.

We find that adversarial examples generated on an undertrained surrogate model transfer significantly more successfully than attacks generated on fully trained models. Our MI-FGSM attacks generated on undertrained models outperform the current state of the art in query-free black-box transfer. Our results suggest that a new focus on finding strong single surrogate models could produce stronger results for adversarial transfer.

We also note that this result has important implications for the analysis of model robustness to black-box attack. Prior work has assumed that surrogate models trained with the same architecture and training procedure are the worst-case for adversarial transfer [19]. However, our results show that this is not the worst-case. An undertrained surrogate model—even one with a different architecture—can produce attacks which transfer more successfully than those based on fully trained models of the same architecture. This suggests that prior robustness analyses underestimate the risk of black-box transfer attacks.

Our work also finds that this effect is not present in robust models, though it is present to some degree in non-robust adversarially trained models, confirming the results given by [28]. Further work here may provide insights into how the training process of robust models differs from that of non-robust models.

Our results also show an interesting contrast between natural transfer and non-robust adversarially trained transfer. As shown in Figure 4.1, naturally trained models show a rapid increase in transferability up to a local maximum at epoch 32, followed by a slow, consistent decline to the final epoch at epoch 90. The maximum epoch for transferability also appears to be associated with the first learning rate decay step for our chosen surrogate model and training parameters. Non-robust adversarially trained models, shown in Figure 4.5b, show a distinctly different pattern of slow, consistent increase in transferability up to a maximum, followed by an extremely rapid reduction in transferability. Intriguingly, models trained using FGSM training exhibit unexpected behavior for iterative attacks. Iterative transfer rates run in lockstep with FGSM transfer rates up to the epoch of maximum transferability, and follow the same pattern of a rapid reduction in transferability over one to two epochs. Then, however, iterative attack transferability slowly increases again, returning to approximately the same level as in the local maximum by the final training epoch. This rapid divergence in transferability between FGSM and I-FGSM examples suggests that non-robust adversarially trained models do not gradually learn resilience to adversarial examples, but do so rapidly at a sudden transition point. Investigating this effect further may reveal stronger approaches to adversarial training or stronger methods for evading adversarially trained models.

Chapter 7

Conclusion and Future Work

Our results show that a simple approach focused on surrogate model, rather than attack method, outperforms standard methods for producing transferable adversarial attacks. We show that this surrogate-focused approach to adversarial example generation creates attacks which transfer well across architectures and models while requiring no query access to the target model. The undertrained surrogate attack outperforms the prior state-of-the-art, ILA-enhanced attacks, by seven to ten percentage points, reducing target classifier performance to below random chance accuracy.

We note that our findings indicate a gap in existing understanding of both adversarial transferability and intermediate epoch models, and that surrogate model investigation represents an open area of investigation for improvements in transfer attacks. Our findings also reveal that the previous known worst-case scenario for black-box transfer (a surrogate model with the same architecture and training procedure) is not an accurate representation of the worst-case, and produces misleading estimates of model robustness to black-box transfer attacks. Evaluation of strategies for producing strong surrogates may provide more insight into the mechanics of transferability, and a more realistic evaluation of the strength of black-box attacks and the robustness of target models.

Our findings leave open many avenues for future work. First among these is how other choices of surrogate model architecture, regularization, and hyperparameters can impact adversarial transferability. Of the surrogate models we evaluate, ResNet18 produces the strongest transfer to the chosen target models. We suggest in Section 5 that this is due to the model's low complexity compared to the other models evaluated. However, it is likely that a more effective surrogate architecture or training method exists. Work on this front would help to identify architectural attributes which produce better gradient similarity and reduced loss curvature. Finally, we suggest that extension of this analysis to different datasets and tasks (such as object detection or semantic segmentation) may provide context for how widespread this effect is.

Section 7.1 Broader Impact

Negative implications of this work include the potential for bad actors to use a surrogate-focused approach to improve their ability to create black-box attacks. Positive implications include increased awareness of the risks of black box attacks, reinforcing the idea that keeping a model's parameters and architecture secure are not sufficient for model security. The understanding that surrogate-focused approaches can outperform other approaches for transfer attacks may lead to increases in understanding of how and why adversarial examples transfer between classifiers. To the best of our knowledge, this method does not leverage any data biases, and we are not aware of any potential consequences of the failure of this approach.

Bibliography

[1] Arjun Nitin Bhagoji, Warren He, Bo Li, and Dawn Song, *Practical black-box attacks on deep neural networks using efficient query mechanisms*, European Conference on Computer Vision, 2018.

[2] Minhao Cheng, Thong Le, Pin-Yu Chen, Huan Zhang, Jinfeng Yi, and Cho-Jui Hsieh, *Query-efficient hard-label black-box attack: An optimization-based approach*, International Conference on Learning Representations, 2019.

[3] Shuyu Cheng, Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu, *Improving black-box adversarial attacks with a transfer-based prior*, Advances in Neural Information Processing Systems, 2019, pp. 10932–10942.

[4] Ambra Demontis, Marco Melis, Maura Pintor, Matthew Jagielski, Battista Biggio, Alina Oprea, Cristina Nita-Rotaru, and Fabio Roli, *Why do adversarial attacks transfer? explaining transferability of evasion and poisoning attacks*, USENIX Security Symposium, 2019.

[5] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li, *Boosting adversarial attacks with momentum*, IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 9185–9193.

[6] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song, *Robust physical-*

world attacks on deep learning visual classification, IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[7] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy, *Explaining and harnessing adversarial examples*, International Conference on Learning Representations (2015).

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, *Deep residual learning for image recognition*, IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[9] Jie Hu, Li Shen, and Gang Sun, *Squeeze-and-excitation networks*, IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[10] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger, *Densely connected convolutional networks*, IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4700–4708.

[11] Qian Huang, Isay Katsman, Horace He, Zeqi Gu, Serge Belongie, and Ser-Nam Lim, *Enhancing adversarial example transferability with an intermediate level attack*, IEEE International Conference on Computer Vision, 2019.

[12] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin, *Black-box adversarial attacks with limited queries and information*, International Conference on Machine Learning, vol. 80, 2018.

[13] Andrew Ilyas, Logan Engstrom, and Aleksander Madry, *Prior convictions: Black-box adversarial attacks with bandits and priors*, International Conference on Learning Representations, 2019.

[14] Alex Krizhevsky, Geoffrey Hinton, et al., *Learning multiple layers of features from tiny images*, (2009).

[15] Alexey Kurakin, Ian Goodfellow, and Samy Bengio, *Adversarial examples in the physical world*, International Conference on Learning Representations, 2016.

[16] Bo Liu, *Geometry and topology of deep neural networks' decision boundaries*, arXiv preprint arXiv:2003.03687 (2020).

[17] Kuang Liu, *Pytorch cifar10 training*, 2018.

[18] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song, *Delving into transferable adversarial examples and black-box attacks*, International Conference on Learning Representations, 2017.

[19] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu, *Towards deep learning models resistant to adversarial attacks*, International Conference on Learning Representations, 2018.

[20] Seungyong Moon, Gaon An, and Hyun Oh Song, *Parsimonious black-box adversarial attacks via efficient combinatorial optimization*, International Conference on Machine Learning, 2019, pp. 4636–4645.

[21] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Jonathan Uesato, and Pascal Frossard, *Robustness via curvature regularization, and vice versa*, IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 9078–9086.

[22] Nicolas Papernot, Ian Goodfellow, Ryan Sheatsley, Reuben Feinman, and Patrick McDaniel, *cleverhans v1.0.0: an adversarial machine learning library*, arXiv preprint arXiv:1610.00768 (2016).

[23] Nataniel Ruiz and Stan Sclaroff, *Disrupting deepfakes: Adversarial attacks against conditional image translation networks and facial manipulation systems*, arXiv preprint arXiv:2003.01279 (2020).

[24] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen, *Mobilenetv2: Inverted residuals and linear bottlenecks*, IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[25] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, *Going deeper with convolutions*, IEEE Conference on Computer Vision and Pattern Recognition, 2015.

[26] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus, *Intriguing properties of neural networks*, International Conference on Learning Representations, 2014.

[27] Chun-Chen Tu, Paishun Ting, Pin-Yu Chen, Sijia Liu, Huan Zhang, Jinfeng Yi, Cho-Jui Hsieh, and Shin-Ming Cheng, *Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks*, AAAI Conference on Artificial Intelligence, vol. 33, 2019, pp. 742–749.

[28] BS Vivek, Konda Reddy Mopuri, and R Venkatesh Babu, *Gray-box adversarial training*, European Conference on Computer Vision, 2018.

[29] Eric Wong, Leslie Rice, and J Zico Kolter, *Fast is better than free: Revisiting adversarial training*, International Conference on Learning Representations, 2019.

[30] Lei Wu, Zhanxing Zhu, Cheng Tai, and Weinan E, *Understanding and enhancing the transferability of adversarial examples*, arXiv preprint arXiv:1802.09707 (2018).

[31] Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille, *Improving transferability of adversarial examples with input*

diversity, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 2730–2739.

[32] Wen Zhou, Xin Hou, Yongjun Chen, Mengyun Tang, Xiangqi Huang, Xiang Gan, and Yong Yang, *Transferable adversarial perturbations*, European Conference on Computer Vision, 2018.