BACKPROPAGATION PATH SEARCH ON ADVERSARIAL TRANSFERABILITY

Anonymous authors
Paper under double-blind review

ABSTRACT

Transfer-based attackers craft adversarial examples against surrogate models and transfer them to victim models deployed in the black-box situation. It is generally accepted that gradients from diverse modules of surrogate models used for perturbation generation contribute differently to transferability. In this paper, we propose backPropagation pAth Search (PAS), which enhances adversarial transferability from the backpropagation perspective. We use structural reparameterization to make the basic modules of DNNs (i.e., convolution and activation) calculate forward as normal but backpropagate the gradients in a skip connection form. Thus, a DAG-based search space is constructed for the backpropagation path. PAS employs Bayesian Optimization to search for the most transferable path and reduces the search overhead by the one-step approximation. We conduct comprehensive attack experiments in a wide range of transfer settings, showing that PAS improves the attack success rate by a huge margin for both normally trained and defense models.

1 INTRODUCTION

Deep neural networks (DNNs) are vulnerable to adversarial examples (Szegedy et al., 2013) despite their success in a wide variety of applications (He et al., 2016; Guo et al., 2017; Kenton & Toutanova, 2019). It is imperative to devise effective attackers to identify the deficiencies of DNNs beforehand, which serves as the first step to improve the model's robustness. White-box attackers (Madry et al., 2018; Carlini & Wagner, 2017; Croce & Hein, 2020) have complete access to the structures and parameters of victim models and effectively cause them to misclassify. However, DNNs are generally deployed in the black-box situation. Transfer-based attackers, as a typical black-box attackers scheme without access to information about the victim model, have drawn more and more attention in the research community (Liu et al., 2017; Xie et al., 2019; Zhang et al., 2022).

Goodfellow et al. (2015) points out that due to the linear nature of DNNs, adversarial examples crafted following a white-box situation against a surrogate model are transferable to unaccessible victim models. To boost adversarial transferability, various methods have been proposed, which focus on different aspects, e.g., momentum terms (Dong et al., 2018; Lin et al., 2019), data augmentation (Xie et al., 2019; Dong et al., 2019), model augmentation (Liu et al., 2017; Li et al., 2020), intermediate features (Ganeshan et al., 2019; Zhang et al., 2022) and meta learning (Fang et al., 2022; Zhu et al., 2021). In this paper, we enhance adversarial transferability from the backpropagation perspective.

Specific network modules (e.g., the widely-used residual module) help with training but do not increase model capacity. Similarly, the backpropagation paths for the gradient of modules have different effects on adversarial transferability. SGM (Wu et al., 2019) and LinBP (Guo et al., 2020) boost adversarial transferability by skipping the gradient from residual modules and nonlinear activation modules, respectively. However, skipping less transferable gradients is limited to the specific modules. As a basic module, convolution modules are used to extract diverse features. Since the most critical features are shared among different DNNs, convolution modules likewise play a key role in adversarial transferability. Besides existing attackers to the intermediate features, the *skip* of less critical convolution modules in backpropagation is missing and worth exploring. Moreover, most existing works are designed in a heuristic manner. Even for boosting transferability via meta learning, the absence of optimizable variables limits its further development.

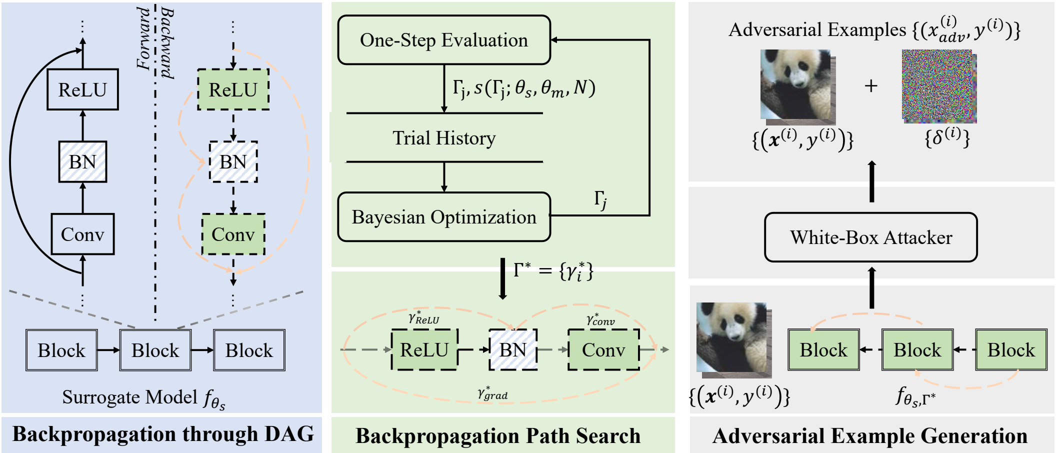

Forward
Backward
ReLU
BN
Conv
ReLU
BN
Conv
Block
Block
Block
Surrogate Model f_{θ_s}
Backpropagation through DAG
One-Step Evaluation
$\Gamma_j, s(\Gamma_j; \theta_s, \theta_m, N)$
Trial History
Bayesian Optimization
Γ_j
$\Gamma^* = \{\gamma_i^*\}$
γ^*_{ReLU}
γ^*_{conv}
γ^*_{grad}
ReLU
BN
Conv
Backpropagation Path Search
Adversarial Examples $\{(x_{adv}^{(i)}, y^{(i)})\}$
+
$\{(\boldsymbol{x}^{(i)}, y^{(i)})\}$
$\{\delta^{(i)}\}$
White-Box Attacker
Block
Block
Block
$\{(\boldsymbol{x}^{(i)}, y^{(i)})\}$
f_{θ_s,Γ^*}
Adversarial Example Generation

Figure 1: Overview of PAS. The blue and green boxes indicate the normal and structural reparameterized modules, respectively. The solid and dashed arrows indicate the forward calculation and backward gradient propagation of models, respectively. After searching for the most transferable path on DAG, all adversarial examples are crafted against the structural reparameterized surrogate.

In this paper, we propose backPropagation pAth Search (PAS), which expands the backpropagation as a Directed Acyclic graph (DAG) and search for the most transferable path. As depicted in Figure 1, PAS craft adversarial examples against the reparameterized surrogate model after the most transferable path on DAG is searched. First, inspired by structural reparameterization for training in RepVGG (Ding et al., 2021), we keep the convolution modules calculating forward as normal but backpropagating the loss in a skip connection form. Specifically, as shown in Figure 2, we reparameterize a single convolution kernel into the sum of two kernels according to the distributivity of convolution. Thus, the reparameterized convolution behaves like a residual module with a skip connection, namely SkipConv. Second, combining the skip paths of convolution, activation, and residual modules in DNNs, we construct a DAG for backpropagation. Figure 3 shows that the gradient is backward propagated from the loss to the input through such DAG by the chain rule. Third, we employ Bayesian Optimization to search for the most transferable path on DAG. Based on the intuitive idea that a highly transferable path attacks any victim model with a high success rate, we adopt an approximation schema to efficiently evaluate the paths and reduce the search overhead. Finally, we generate adversarial examples against the surrogate model based on the most transferable backpropagation path. Extensive experiments on the subsets of ImageNet from different surrogate models demonstrate the effectiveness of PAS against both normally trained and defense models in comparison with the baseline and state-of-the-art (SOTA) transfer-based attackers.

Our main contributions can be summarized as follows:

- We propose SkipConv, which acts as standard convolutions during the forward phase but backpropagates loss in a skip connection form via structural reparameterization. We further propose a DAG-based search space for the backpropagation path by combining the existing structural reparameterization of residual and activation modules.
- To our best knowledge, we propose the first transfer-based attacker to search the backpropagation path for adversarial transferability. PAS employs Bayesian Optimization to search for the most transferable path and reduces the search overhead by one-step approximation.
- We conduct comprehensive transfer attack experiments in a wide range of transfer settings, showing that PAS improves the attack success rate by a huge margin for both normally trained and defense models.

2 Preliminary

Given a clean example $\boldsymbol{x}$ with class label y and a victim model f_θ parameterized by θ, the goal of an adversary is to find an adversarial example $\boldsymbol{x}_{adv}$, which is constrained by L_p norm with a bound

ϵ, to fool the model into making an incorrect prediction:

$$f_\theta(\boldsymbol{x}_{adv}) \neq y, \text{ where } \|\boldsymbol{x}_{adv} - \boldsymbol{x}\|_p \leq \epsilon \tag{1}$$

In the white-box situation, FGSM (Goodfellow et al., 2015) perturbs the clean example $\boldsymbol{x}$ for one step by the amount of ϵ along the gradient direction. As an iterative version, I-FGSM (Kurakin et al., 2018b) perturbs $\boldsymbol{x}$ for T steps with smaller step size η and achieves a high attack success rate:

$$\boldsymbol{x}_{adv}^{t+1} = \Pi_\epsilon^{\boldsymbol{x}} \left(\boldsymbol{x}_{adv}^t + \eta \cdot \text{sign}\left(\nabla_{\boldsymbol{x}} l \left(f_\theta \left(\boldsymbol{x}_{adv}^t \right), y \right) \right) \right), \text{ where } t \in \{0, \ldots, T-1\} \tag{2}$$

Without access to the victim model f_θ, transfer-based attackers craft adversarial examples against a white-box surrogate model $f_{\theta_s, \Gamma}$ with structure hyper-parameters Γ (e.g., hyper-parameters for residual and activation modules) to achieve Equation 1:

$$\boldsymbol{x}_{adv}^{t+1} = \Pi_\epsilon^{\boldsymbol{x}} \left(\boldsymbol{x}_{adv}^t + \eta \cdot \text{sign}\left(\nabla_{\boldsymbol{x}} l \left(f_{\theta_s, \Gamma} \left(\boldsymbol{x}_{adv}^t \right), y \right) \right) \right), \text{ where } t \in \{0, \ldots, T-1\} \tag{3}$$

Backpropagation is essential in the process of adversarial example generation. Classical DNNs is consisted of several layers, i.e., $f = f_1 \circ \cdots \circ f_L$, where $i \in \{1, \ldots L\}$ is the layer index, and $\boldsymbol{z}_i = f_i(\boldsymbol{z}_{i-1})$ indicates the intermediate output and $\boldsymbol{z}_0 = \boldsymbol{x}$. According to the chain rule in calculus, the gradient of the loss l with respect to input $\boldsymbol{x}$ can be then decomposed as:

$$\frac{\partial l}{\partial \boldsymbol{x}} = \frac{\partial l}{\partial \boldsymbol{z}_L} \frac{\partial f_L}{\partial \boldsymbol{z}_{L-1}} \cdots \frac{\partial f_1}{\partial \boldsymbol{z}_0} \frac{\partial \boldsymbol{z}_0}{\partial \boldsymbol{x}} \tag{4}$$

Thus, a single path is used for the gradient propagation backward from the loss to the input. Taking the common ReLU activation module f_i^{ReLU} as an example, the gradient is propagated backward as $\partial f_i^{ReLU} / \partial \boldsymbol{z}_{i-1} = W_i M_i W_{i+1}$, where M_i is a diagonal matrix whose entries are 1 if the corresponding entris of $W_i^T \boldsymbol{z}_{i-1}$ are positive and 0 otherwise. Extending f to a ResNet-like (with skip connections) networks, the residual module in layer i where $f_i^{res}(\boldsymbol{z}_{i-1}) = \boldsymbol{z}_{i-1} + f_i(\boldsymbol{z}_{i-1})$ decomposes the gradient as:

$$\frac{\partial l}{\partial \boldsymbol{z}_0} = \frac{\partial l}{\partial \boldsymbol{z}_L} \cdots \frac{\partial f_i^{res}}{\partial \boldsymbol{z}_{i-1}} \cdots \frac{\partial \boldsymbol{z}_0}{\partial x} = \frac{\partial l}{\partial \boldsymbol{z}_L} \cdots \left(1 + \frac{\partial f_i}{\partial \boldsymbol{z}_{i-1}} \right) \cdots \frac{\partial \boldsymbol{z}_0}{\partial x} \tag{5}$$

Such residual module provides a gradient highway for training (He et al., 2016). Similarly, the use of skip connections in backpropagation allows easier generation of highly transferable adversarial examples. SGM (Wu et al., 2019) introduces a decay parameter to use more gradients from the skip connections in residual modules, i.e., $\partial f_i^{res} / \partial \boldsymbol{z}_{i-1} = 1 + \gamma \cdot \partial f_i / \partial \boldsymbol{z}_{i-1}$. LinBP (Guo et al., 2020) skips the ReLU module and renormalizes the gradient passing backward as $\partial f_i^{ReLU} / \partial \boldsymbol{z}_{i-1} = \alpha_i \cdot W_i W_{i-1}$ where $\alpha_i = \|W_i M_i W_{i-1}\|_2 / \|W_i W_{i-1}\|_2$.

3 Methodology

In this section, we first introduce how we use structural reparameterization to expand the backpropagation as a DAG in Section 3.1. Then, to reduce the search overhead, we propose a one-step approximation in Section 3.2. Finally, we present the overall process of PAS in Section 3.3.

3.1 Backpropagation DAG

In this part, we introduce how we expand the backpropagation as a DAG via structural reparameterization. Unlike works that use the existing skip connections of the surrogate model (e.g., residual module), PAS reparameterizes normal modules with skip connections to search for transferable backpropagation paths.

Skip Convolution. As a basic module, convolution extracts diverse features and affects adversarial transferability since the most critical features are shared among different DNNs. Unlike existing attackers to intermediate features (Ganeshan et al., 2019; Zhang et al., 2022), we propose SkipConv and use more gradients from critical features through skip connection.

SkipConv is realized by structural reparameterization. As shown in the Figure 2, for a convolution module f_i^{conv} with kernel k_i, we expand the kernel according to convolution distributivity as the sum

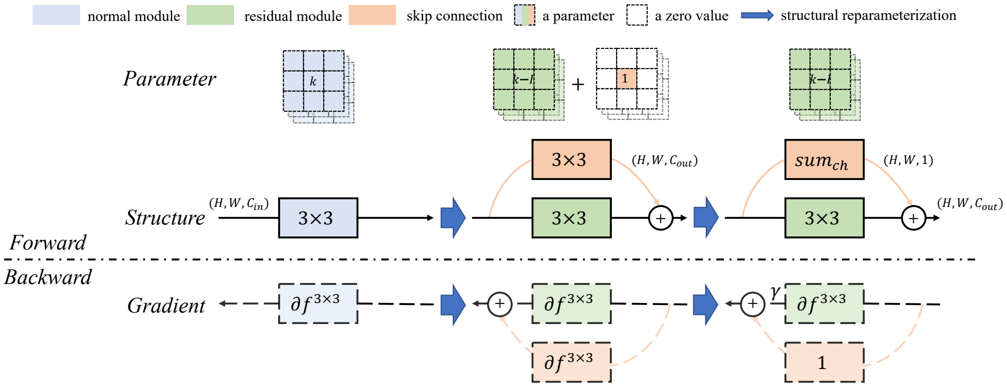


Figure 2: SkipConv: structural reparameterization of convolution module. Take the 3×3 convolution as an example. According to convolution distributivity, the normal kernel k is reparameterized into the sum of the skip kernel and I the residual kernel $k - I$. γ is introduced to control the gradient from the residual kernel.

of a constant 1×1 kernel I with all values 1 and the corresponding residual kernel $k_i - I$. Since all values of the kernel I are 1, we replace the special convolution with the sum of each channel as a skip connection. In this way, we reparameterize the normal convolution into the skip connection I and the residual $k_i - I$. A decay factor $\gamma_i \in [0, 1]$ is introduced as the weight of the residual gradient in backpropagation. Thus, we modify the convolution module in a skip connection form as:

$$f_i^{conv}(\boldsymbol{z}_{i-1}; k_i) = sum_{ch}(\boldsymbol{z}_{i-1}) + \gamma_i \cdot f_i^{conv}(\boldsymbol{z}_{i-1}; k_i - I) + C \tag{6}$$

where C is equal to $(1 - \gamma_i) \cdot f_i^{conv}(\boldsymbol{z}_{i-1}; k_i - I)$ without gradient backward. Such SkipConv requires no fine-tuning since it calculates forward as normal. For backpropagation, γ_i is used to relatively adjust the gradient of the residual, i.e., $1 + \gamma_i \cdot \partial f_i^{conv}(\boldsymbol{z}_{i-1}; k_i - I)/\partial \boldsymbol{z}_{i-1}$.

Skip Activation. ReLU is a common activation module in neural networks. Guo et al. (2020) demonstrates that the gradient of ReLU is sparse, which degrades adversarial transferability. LinBP skips the gradient of ReLU with a dense all-1 matrix and then normalizes the gradient. However, the scalar α_i used for normalization needs to be calculated based on the weight of the front and back layers. We further devise an approximation for α_i and reparameterize ReLU as follows:

$$f_i^{ReLU}(\boldsymbol{z}_{i-1}) = \hat{\alpha}_i \cdot (\boldsymbol{z}_{i-1} + ReLU(-\boldsymbol{z}_{i-1})) + (1 - \hat{\alpha}_i) \cdot ReLU(\boldsymbol{z}_{i-1}) \tag{7}$$

where $\hat{\alpha}_i = \|M_i\|_2 / \|z_{i-1}\|_2$ uses the sparsity as the estimation of the re-normalizing factor.

Skip Gradient. For gradient paths within different layers, we use the following SkipGrad in SGM:

$$f_i^{res}(\boldsymbol{z}_{i-1}) = \boldsymbol{z}_{i-1} + \gamma_i \cdot f_i(\boldsymbol{z}_{i-1}) + C \tag{8}$$

where C is equal to $(1 - \gamma_i) \cdot f_i(\boldsymbol{z}_{i-1})$ without gradient backward.

In summary, we reparameterize the structure of diverse basic modules in DNNs and control the weight of backpropagation paths by γ. For each module's backpropagation path, we control the gradient backward via SkipConv and LinReLU. For cross-module paths, we use the existing skip connection as a highway for adversarial transferability. By combining all the paths of the above modules, we construct the Directed Acyclic Graph (DAG) for gradient propagation backward. As shown in Figure 3, we use $\Gamma = \{\gamma_i\}$ to control the weight of the residual path, and hence black-box optimization can be used to search for the most transferable paths.

3.2 One-Step Approximation for Path Evaluation

In this part, we show how PAS efficiently evaluates the adversarial transferability of backpropagation paths.

To guide the search on the backpropagation DAG, we need to evaluate the sampled paths. Different from other attackers (Yuan et al., 2021; Zhu et al., 2021), we propose the one-step approximation to

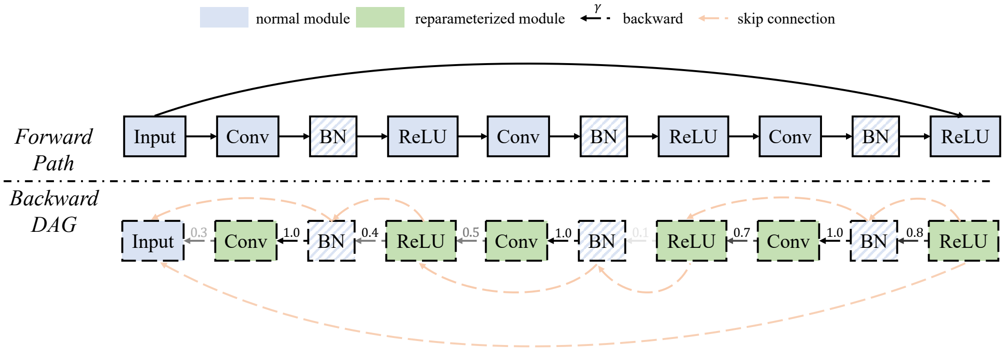


Figure 3: Example of backpropagation DAG. The color transparency indicates the weight γ of the corresponding path.

alleviate the large overhead consumed in the search process. It is a good and intuitive rule of thumb that the highly transferable paths have a high attack success rate on all data for any victim model. Based on this, an approximate schema is adopted, i.e., we use the one-step attack success rate of samples on only one white-box meta victim model as the estimation of transferability.

To verify the improved transferability, we leverage the chi-square test to judge whether there is a significant difference between candidates. Specifically, the question is whether there is a significant difference in the transferability of paths p_1 and p_2 on a meta dataset of N samples, for which the attack success rate is s_1 and s_2, respectively. We calculate the statistic as $stats = \frac{2N(s_1-s_2)^2}{(s_1+s_2)(2-s_1-s_2)}$, which follows the χ^2 distribution with 1 degree of freedom. The corresponding statistics should satisfy $stats \geq 3.841$ for the confidence interval of 95%. Thus, we maintain a sliding window to store the paths that are not significantly different as candidates in the search process based on statistical confidence.

All in all, to efficiently evaluate the transferability of paths, we use only a vgg19 as the meta victim model θ_m, and randomly sample $N = 200$ clean examples as the meta dataset to calculate the one-step attack success rate as Equation 9. Then, the paths without significant differences are stored as candidates for further evaluation.

$$s(\Gamma; \theta_s, \theta_m, N) = \frac{1}{N}\sum_{i=0}^{N} \mathbf{1}\left(f_{\theta_m}\left(\Pi_{\epsilon}^{\boldsymbol{x}^{(i)}}\left(\boldsymbol{x}^{(i)} + \eta \cdot \text{sign}\left(\nabla_{\boldsymbol{x}^{(i)}} l\left(f_{\theta_s,\Gamma}(\boldsymbol{x}^{(i)}), y^{(i)}\right)\right)\right)\right) \neq y^{(i)}\right) \tag{9}$$

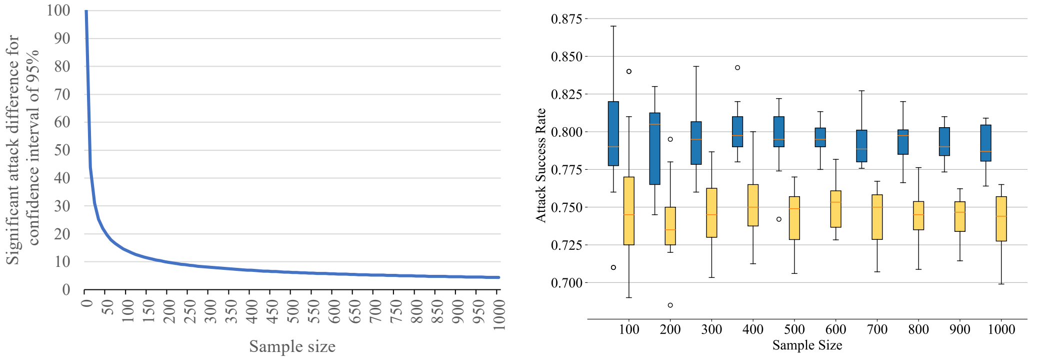
Considering the worst case (i.e., $s_1 + s_2 = 1$), we plot the relationship between the sample size N and the significant difference of attack success rate $|s_1 - s_2|$. As shown in Figure 4(a), we need 200 samples (i.e., 20% of the test set) to observe better paths with a 10% difference in the attack success rate. To further verify its effectiveness, we select two paths with a difference of 5% in the test set, sample meta datasets of different sizes 20 times, and draw a box plot. Figure 4(b) shows that the difference in transferability is more obvious as the size of the meta dataset increases. Moreover, the evaluation of adversarial transferability on 200$\sim$300 samples is sufficient to distinguish between the two paths, except for a few outliers.

3.3 PAS: SEARCH FOR ADVERSARIAL TRANSFERABILITY

In this part, we introduce how PAS searches the backpropagation DAG for adversarial transferability.

To optimize the above objective, we use Bayesian optimization[1] to search the structure parameters Γ and combine it with Hyperband (Li et al., 2017) to allocate resources for each trial of the sampled path. The overall procedure is shown in Algorithm 1. We first search for the most transferable path

[1] https://optuna.org/



(a) Significant difference of attack success rate in the worst case.

(b) Statistics of attack success rates at different sample sizes for paths with $|s_1 - s_2| = 5\%$ on the test set.

Figure 4: Relationship between sample size N and difference of attack success rate $|s_1 - s_2|$.

Γ^* of the surrogate model according to Equation 9 and then craft adversarial examples, which are transferred to unaccessible victim models.

In the search process, PAS reparameterizes the structure of the surrogate model and initializes Γ. Bayesian Optimization is used to sample the backpropagation path Γ_k. According to the sampled paths, adversarial examples for the meta dataset are crafted against f_{θ_s,Γ_k}. PAS calculates the attack success rate on the meta victim model and uses it as the feedback for Bayesian Optimization for the next iteration. When predefined resources are exhausted, PAS uses the optimal structure Γ^* to craft adversarial examples on the test set and transfers them to all victim models.

Algorithm 1 PAS: Backpropagation Path Search on Adversarial Transferability

Input: Surrogate model f_{θ_s}, meta victim model θ_m, perturbation bound ϵ, the number of attack steps T, the number of trials N_t

```
Reparameterize the structure of $f_{\theta_s}$ as $f_{\theta_s,\Gamma}$
for $j = 1, \ldots, N_t$ do
    sample $\Gamma_j$ by Bayesian Optimization according to the trail history
    evaluate $\Gamma_j$ by Equation 9 and add it to the history
end for
select the most transferable path $\Gamma^*$ according to the history
return adversarial examples crafted against $f_{\theta_s,\Gamma^*}$
```

4 Experiments

In this section, we conduct extensive experiments to investigate the effectiveness of PAS.

4.1 Experiment Setup

Dataset. To compare with baselines, we report the results on two datasets: 1) Subset1000: ImageNet-compatible dataset in the NIPS 2017 adversarial competition (Kurakin et al., 2018a), which contains 1000 images; 2) Subset5000: a subset of ImageNet validation images, which contains 5000 images and is used by SGM and IAA. We check that all of the models are almost approaching 100% classification success rate in this paper.

Models. We conduct experiments on both normally trained models and defense models. For normal trained models, we consider 7 models containing VGG19 (Simonyan & Zisserman, 2014), ResNet-152 (RN152) (He et al., 2016), DenseNet-201 (DN201) (Huang et al., 2017), Squeeze-and-Excitation network (SE154) (Hu et al., 2018), Inception-v3 (IncV3) (Szegedy et al., 2016), Inception-v4 (IncV4) and Inception-Resnet-v2 (IncRes) (Szegedy et al., 2017). For defense models, we select advanced defense methods covering random resizing and padding (R&P) (Xie et al., 2018), NIPS-r3[2], randomized smoothing (RS) (Cohen et al., 2019), and 3 robustly trained models using ensemble adversarial training (Tramèr et al., 2018): ensemble of 3 IncV3 models (IncV3$_{\text{ens3}}$), ensemble of 4 IncV3 models (IncV3$_{\text{ens4}}$) and ensemble of 3 IncResV2 models (IncResV2$_{\text{ens3}}$). We choose different models (i.e., RN152, DN201, RN50, RN121, IncV4, and IncResV2) as surrogate models to compare with different baselines. As PAS is inspired by RepVGG, VGG19 is used as the meta model to evaluate the transferability of backpropagation paths.

Baseline Methods. To demonstrate the effectiveness of PAS, we compare it with existing competitive baselines, i.e., I-FGSM (Kurakin et al., 2018b), MI (Dong et al., 2018), DI (Xie et al., 2019), SGM (Wu et al., 2019), LinBP (Guo et al., 2020), IAA (Zhu et al., 2021), LLTA (Fang et al., 2022), FDA (Ganeshan et al., 2019), FIA (Wang et al., 2021) and NAA (Zhang et al., 2022).

Metrics. Following the most widely adopted setting, we use the attack success rate as the metric. Specifically, the attack success rate is defined as the percentage of adversarial examples that successfully mislead the victim model among all adversarial examples generated by the attacker.

Hyperparameter. For the search process in PAS, we conduct $K = 2000$ trials to search on the backpropagation DAG for each surrogate model, which evaluates the transferability of the backpropagation path on 256 examples in one-step attacks against the meta model (i.e., VGG19). The overall search overhead is approximately 20 times that of generating adversarial samples in a 10-step attack on the test set. To craft adversarial examples, we use the hyperparameter setting in Zhang et al. (2022) to set the maximum perturbation of $\epsilon = 16$, the number of attack steps $T = 10$ and the step size $\eta = 1.6/255$. Moreover, for a fair comparison on Subset5000 with IAA, which is not open source, we follow its parameter setting and set the step size to $\eta = 2/255$.

4.2 Attack Normally Trained Models (RQ1)

Table 1: Attack success rate (%) against normally trained and defense models on Subset5000. The best results are in **bold**.

	Attacker	RN152	DN201	SE154	IncV3	IncV4	IncRes	IncV3$_{\text{ens3}}$	IncV3$_{\text{ens4}}$	IncRes$_{\text{ens3}}$
RN152	I-FGSM	99.91	51.00	26.32	23.50	22.58	18.72	12.20	10.80	5.70
	MI	99.82	75.79	53.00	46.50	43.32	33.08	24.20	22.04	16.10
	DI	99.78	77.81	57.49	50.28	47.16	35.10	35.97	32.81	20.16
	SGM	99.87	82.76	61.90	53.16	49.24	43.30	31.57	27.77	20.84
	IAA	99.87	95.06	82.46	76.34	71.04	58.34	43.28	37.88	26.78
	PAS	**99.96**	**96.76**	**84.98**	**83.82**	**78.82**	**77.18**	**59.34**	**54.46**	**44.74**
DN201	I-FGSM	59.08	99.89	40.60	33.80	32.46	23.80	18.16	15.30	10.40
	MI	76.39	99.84	64.38	59.62	54.85	39.40	31.79	28.21	20.60
	DI	78.18	99.81	61.75	60.04	56.15	40.56	42.76	42.01	34.28
	SGM	86.60	99.67	72.20	62.34	56.36	45.42	41.45	37.85	29.41
	IAA	93.82	**99.78**	87.98	88.26	87.02	79.12	61.02	53.80	46.34
	PAS	**96.06**	99.76	**90.94**	**91.00**	**88.12**	**85.96**	**75.08**	**72.22**	**62.28**

In this part, we investigate the transferability of attackers against normally trained models.

We report the attack success rates of PAS, baselines and backpropagation-based attackers with RN152 and DN121 as the surrogate model on Subset5000 in Table 1. Table 1 demonstrates that PAS beats other attackers in all black-box scenarios. Averagely, PAS achieves 88.13% attack success rates for RN152, which is 5.62% higher than IAA and 20.90% higher than SGM. For DN201, PAS achieves an average improvement of 2.25% in comparison with IAA, and we observe a better improvement for PAS in victim models, which are more difficult to attack (e.g., 6.84% improvement against IncRes). Since SGM manually tunes the decay factors for SkipGrad and IAA uses Bayesian optimization for SkipGrad and LinReLU, we owe the improvement to both the DAG search space and the efficient one-step approximation of PAS, which boosts adversarial transferability.

Table 2: Attack success rate (%) against robustly trained models on Subset1000. The best results are in **bold**.

	Attacker	IncV3$_{\text{ens3}}$	IncV3$_{\text{ens4}}$	IncRes$_{\text{ens3}}$		Attacker	IncV3$_{\text{ens3}}$	IncV3$_{\text{ens4}}$	IncRes$_{\text{ens3}}$
RN50	I-FGSM	17.3	18.5	11.2	IncV4	MI-PD	23.9	24.5	12.5
	SGM	30.4	28.4	18.6		FDA-MI-PD	21.9	20.9	9.1
	LinBP	34.5	32.5	20.9		FIA-MI-PD	45.5	42.1	23.5
	LLTA	50.6	47.3	33.6		NAA-MI-PD	55.4	53.6	34.4
	PAS	**72.8**	**70.4**	**57.9**		PAS-MI-DI	**71.5**	**66.8**	**49.7**
DN121	I-FGSM	21.8	21.5	13.1	IncRes	MI-PD	28.8	26.7	16.3
	SGM	36.8	36.8	22.5		FDA-MI-PD	17.4	29.9	25.3
	LinBP	39.3	38.3	22.6		FIA-MI-PD	49.7	44.9	31.9
	LLTA	59.1	60.5	46.8		NAA-MI-PD	61.9	59.0	48.3
	PAS	**70.9**	**70.8**	**57.4**		PAS-MI-DI	**76.9**	**71.2**	**59.8**

4.3 Attack Defense Models (RQ2)

In this part, to further verify the superiority of PAS, we conduct a series of experiments against defense models. We illustrate the attacking results against competitive baseline methods under various experimental settings.

Table 1 shows the attack success rate on Subset5000. The advantages of PAS are more obvious against defense models. The average attack success rates are 52.85% and 69.86% for RN152 and DN201, respectively, which are 16% more than the second-best attacker IAA.

For the commonly used Subset1000, we directly attack defense models since most of the existing attackers have achieved a 90% attack success rate against normally trained models. The comparisons between PAS and the feature-level and backpropgation-based attackers are presented in Table 2. Table 2 demonstrate that highly transferable attacks are crafted against defense models in average of 23.2% and 10.9% by PAS. Although LLTA tunes the data and model augmentation through meta tasks, PAS searches the backpropagation DAG and achieves higher transferability, which shows the improvement that comes with a larger search space.

We further demonstrate that the adversarial transferability of PAS can be exploited in combination with existing methods. In contrast to the results in LLTA that DI conflicts with LinBP and leads to large performance degradation, we combine PAS with DI for transferability gains. As shown in Table 2, when combined with both MI and DI, PAS improves the SOTA transferability by a huge margin consistently against robustly trained models by at least 11.5%. In addition, the attack success rate in Table 3 demonstrates that PAS outperforms SOTA feature-level attackers with an average improvement of 7.7%.

All in all, the experimental results identify higher adversarial transferability of PAS against defense models. Compared with the existing methods, PAS achieves a 6.9%$\sim$24.3% improvement in attack success rate and demonstrates the generality with various surrogate models on two benchmarks.

4.4 Ablation Study

In this part, we conduct the ablation study to verify the contribution of each part in PAS by removing skip modules in DAG and performing hyperparameters experiments.

Skip modules. We utilize PAS on different search spaces to search for the backpropagation path and observe the attack success rate. As mentioned above, DAG consists of three kinds of skip module, i.e., SkipConv, LinReLU and SkipGrad. Hence, we show the attack success rate of DAG without each kind of skip module and DAG with only one kind of skip module, respectively. The experimental results are reported in Table 4, and we draw the following conclusions:

- SkipConv leverages the gradient from the critical features and achieves the highest attack success rate among all DAGs with a single skip module.

[2] https://github.com/anlthms/nips-2017/tree/master/mmd

Table 3: Attack success rate (%) against defense models. Avg. indicates the average success rate on all black-box victim models. The best results are in **bold**.

	R&P	NIPS-r3	RS	Avg.
MI-PD	22.4	28.8	31.4	27.5
FDA-MI-PD	16.3	23.1	27.8	22.4
FIA-MI-PD	36.4	51.2	38.4	42.0
NAA-MI-PD	46.8	62.9	40.4	50.0
PAS-MI-DI	**55.2**	**69.8**	**48.1**	**57.7**

Table 4: The statistics of attack success rate (%): w/o indicates the search space without the skip module; w/ indicates the search space with the skip module.

	Normal	Defense	Total
PAS	90.43	66.63	83.94
w/o SkipConv	77.90	52.30	70.92
w/o SkipGrad	34.30	22.40	31.05
w/o LinReLU	76.35	38.33	65.98
w/ SkipConv	76.16	38.33	65.85
w/ SkipGrad	57.36	23.80	48.21
w/ LinReLU	33.11	20.17	29.58

- Since LinReLU is realized in an approximation form (i.e., $\hat{\alpha}$), adversarial transferability degrades without scaling the gradient by SkipGrad.
- The most transferable path is searched for by combining all skip modules and achieves at least a 13.02% improvement compared with the variants.

Number of instances for evaluation N. We measure the one-step approximation of transferability by altering the sample size for path evaluation. We observe from Figure 4(b) that with the increase of instances for evaluation, the difference in paths is more obvious. However, along with the obvious difference caused by larger N, the overhead linearly increases. To balance the performance and evaluation overhead, we choose $N = 200$ to achieve adequate performance.

5 Related Work

Black-box attackers can be roughly divided into query-based and transfer-based schemes. Query-based attackers estimate gradient with queries of the prediction to the victim model (Papernot et al., 2017; Su et al., 2019). Due to the lack of access to numerous queries in reality, part of query-based attackers focus on improving efficiency and reducing queries. In contrast, transfer-based attackers do not require any query and can be applied to unaccessible victim models.

To boost adversarial transferability, various methods have been proposed: regarding the adversarial example generation as an optimization process, Dong et al. (2018); Lin et al. (2019) leverage momentum terms to escape from poor local optima. To avoid overfitting with the surrogate model and specific data pattern, data augmentation (Xie et al., 2019; Dong et al., 2019) and model augmentation (Liu et al., 2017; Li et al., 2020) are effective strategies. Since the most critical features are shared among different DNNs, feature-level attackers Ganeshan et al. (2019); Zhang et al. (2022) destroy the intermediate feature maps. From the backpropagation perspective, Wu et al. (2019); Guo et al. (2020) leverages more gradients from paths of more useful modules. Unlike most methods which are designed in a heuristic manner, Fang et al. (2022); Yuan et al. (2021); Zhu et al. (2021) enhances adversarial transferability by black-box optimization and meta learning.

6 Conclusion

In this paper, we enhance adversarial transferability from the backpropagation perspective and propose PAS to search backpropagation paths for adversarial transferability. We propose SkipConv, which calculates forward as normal convolution modules but backpropagates loss in a skip connection form through structural reparameterization. We construct a DAG-based search space for the backpropagation path by combining the existing structural reparameterization of residual and activation modules. Then, we employ Bayesian Optimization to search for the most transferable path and further reduce the search overhead by one-step approximation for path evaluation. The results of comprehensive attack experiments in a wide range of transfer settings show that PAS improves the attack success rate by a huge margin for both normally trained and defense models.

REFERENCES

Nicholas Carlini and David A. Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pp. 39–57. IEEE Computer Society, 2017. doi: 10.1109/SP.2017.49. URL https://doi.org/10.1109/SP.2017.49.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, pp. 1310–1320. PMLR, 2019.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, 2020.

Xiaohan Ding, Xiangyu Zhang, Ningning Ma, Jungong Han, Guiguang Ding, and Jian Sun. Repvgg: Making vgg-style convnets great again. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13733–13742, 2021.

Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9185–9193, 2018.

Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 4312–4321. Computer Vision Foundation / IEEE, 2019. doi: 10.1109/CVPR.2019.00444. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Dong_Evading_Defenses_to_Transferable_Adversarial_Examples_by_Translation-Invariant_Attacks_CVPR_2019_paper.html.

Shuman Fang, Jie Li, Xianming Lin, and Rongrong Ji. Learning to learn transferable attack. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 571–579, 2022.

Aditya Ganeshan, B S Vivek, and R. Venkatesh Babu. Fda: Feature disruptive attack. *international conference on computer vision*, 2019.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6572.

Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. Deepfm: a factorization-machine based neural network for ctr prediction. In *Proc. of IJCAI*, 2017.

Yiwen Guo, Qizhang Li, and Hao Chen. Backpropagating linearly improves transferability of adversarial examples. *Advances in Neural Information Processing Systems*, 33:85–95, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7132–7141, 2018.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, 2019.

Alexey Kurakin, Ian Goodfellow, Samy Bengio, Yinpeng Dong, Fangzhou Liao, Ming Liang, Tianyu Pang, Jun Zhu, Xiaolin Hu, Cihang Xie, et al. Adversarial attacks and defences competition. In *The NIPS'17 Competition: Building Intelligent Systems*, pp. 195–231. Springer, 2018a.

Alexey Kurakin, Ian J Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pp. 99–112. Chapman and Hall/CRC, 2018b.

Lisha Li, Kevin Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *The Journal of Machine Learning Research*, 18(1):6765–6816, 2017.

Yingwei Li, Song Bai, Yuyin Zhou, Cihang Xie, Zhishuai Zhang, and Alan Yuille. Learning transferable adversarial examples via ghost networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 11458–11465, 2020.

Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and John E Hopcroft. Nesterov accelerated gradient and scale invariance for adversarial attacks. In *International Conference on Learning Representations*, 2019.

Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL `https://openreview.net/forum?id=Sys6GJqxl`.

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

Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pp. 506–519, 2017.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23(5):828–841, 2019.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-first AAAI conference on artificial intelligence*, 2017.

Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble adversarial training: Attacks and defenses. In *International Conference on Learning Representations*, 2018.

Zhibo Wang, Hengchang Guo, Zhifei Zhang, Wenxin Liu, Zhan Qin, and Kui Ren. Feature importance-aware transferable adversarial attacks. *international conference on computer vision*, 2021.

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 *International Conference on Learning Representations*, 2019.

Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan Yuille. Mitigating adversarial effects through randomization. In *International Conference on Learning Representations*, 2018.

Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2730–2739, 2019.

Zheng Yuan, Jie Zhang, Yunpei Jia, Chuanqi Tan, Tao Xue, and Shiguang Shan. Meta gradient adversarial attack. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7748–7757, 2021.

Jianping Zhang, Weibin Wu, Jen-tse Huang, Yizhan Huang, Wenxuan Wang, Yuxin Su, and Michael R Lyu. Improving adversarial transferability via neuron attribution-based attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14993–15002, 2022.

Yao Zhu, Jiacheng Sun, and Zhenguo Li. Rethinking adversarial transferability from a data distribution perspective. In *International Conference on Learning Representations*, 2021.