Benchmarking Transferable Adversarial Attacks

Zhibo Jin*, Jiayu Zhang†, Zhiyu Zhu*and Huaming Chen*
*The University of Sydney
†Suzhou Yierqi

***Abstract*—The robustness of deep learning models against adversarial attacks remains a pivotal concern. This study presents, for the first time, an exhaustive review of the transferability aspect of adversarial attacks. It systematically categorizes and critically evaluates various methodologies developed to augment the transferability of adversarial attacks. This study encompasses a spectrum of techniques, including Generative Structure, Semantic Similarity, Gradient Editing, Target Modification, and Ensemble Approach. Concurrently, this paper introduces a benchmark framework *TAA-Bench*, integrating ten leading methodologies for adversarial attack transferability, thereby providing a standardized and systematic platform for comparative analysis across diverse model architectures. Through comprehensive scrutiny, we delineate the efficacy and constraints of each method, shedding light on their underlying operational principles and practical utility. This review endeavors to be a quintessential resource for both scholars and practitioners in the field, charting the complex terrain of adversarial transferability and setting a foundation for future explorations in this vital sector. The associated codebase is accessible at: https://github.com/KxPlaug/TAA-Bench**

I. Introduction

In recent years, adversarial attacks have emerged as a significant research direction for artificial intelligence and machine learning, especially in the context of the security of deep learning. It originates from the observation that deep neural networks (DNNs) are sensitive to subtle perturbations in input data. Even imperceptible to the human eye, such changes can lead to incorrect output results [3]. Adversarial attacks can be categorized into two types based on the availability of model data: white-box attacks and black-box attacks [1], [18], [6], [19]. White-box attacks assume the model's internal information are accessible, such as its parameters, structure, and training data. In contrast, black-box attacks occur without knowledge of the internal information of the attacked model, which aligns more closely with real-world scenarios, as attackers often have constraints in access operation.

From the technical perspective, black-box attacks can be classified into two categories: query-based attacks and transferable adversarial attacks. The former, although unable to directly access the model's content, requires multiple requests to the target model [23]. Thus, attackers can infer and construct an approximate model to launch the attack. A limitation is that it necessitates regular and extensive access to the target model, thereby reducing the stealthiness of the attack. On the contrary, our emphasis is on a more practical form of attack, namely transferable adversarial attacks (TAA). This method involves the use of a surrogate model, which, while distinct from the target model, shares similar features or functionalities. Adversarial samples generated on this surrogate model are then used to attack the target model [22]. Owing to certain generalization properties inherent in deep learning models, these adversarial samples often successfully mislead the target model, different from the surrogate, thus facilitating a transfer attack. The key advantage of this method lies in the ability to operate without direct access or querying of the target model, thereby enhancing the stealth and practicality of the attack.

Despite the widespread research on transferable adversarial attacks in recent years, it lacks a comprehensive and systematic retrospective study. Thus, this paper aims to systematically review and categorise the existing classical and latest TAA methods. For the first time, we investigate TAA methods from multiple dimensions, categorising existing approaches into generative architecture, construction of semantic similarity, gradient editing types, modification of attack targets and ensemble types. Additionally, we benchmark baseline methods for comparison. Moreover, we reproduce 10 representative methods of transferable adversarial attacks and integrated these methods into an open-source benchmark framework, which is published on GitHub *TAA-Bench*, facilitating related researches. Our main contributions are:

- We thoroughly collate existing methods of transferable adversarial attacks, and systematically analyse their implementation principles.
- We present an extensible, modular and open-source benchmark *TAA-Bench* that includes implementations of different types of transferable adversarial attacks to facilitate research and development in this field.

II. Problem Definition

In the study of transferability in adversarial attacks, our objective is to generate a slightly perturbed input sample x', causing misclassification of a black-box target deep learning model while keeping these changes imperceptible to human observers. Specifically, consider a surrogate deep learning model f with parameters θ, a black-box target deep learning model f', a representative input sample x along with its corresponding true label y, and a small perturbation magnitude ϵ. Our goal is to find a perturbation δ such that $x' = x + \delta$ satisfies two conditions: 1) the target model f' does not output the true label y when predicting the input sample x', i.e., $f'(x') \neq y$; 2) the magnitude of the perturbation δ is constrained to be within a given threshold ϵ, ensuring that x' remains indistinguishable from original sample x to human.

Workshop on AI Systems with Confidential Computing (AISCC) 2024
26 February 2024, San Diego, CA, USA
ISBN 979-8-9894372-4-5
https://dx.doi.org/10.14722/aiscc.2024.23017
www.ndss-symposium.org

TABLE I. COMPARATIVE ANALYSIS OF ADVERSARIAL ATTACK STRATEGIES

Strategy	Description	Examples	Advantages	Disadvantages
Generative Architecture	Generates perturbations via a dedicated network	[15], [24]	Fast attack execution	Requires extra network training, complex training process
Semantic Similarity	Uses semantically similar samples in attacks	[16], [7], [9], [14], [11], [25]	Simple to deploy and motivate	High-quality sample generation can be challenging
Gradient Editing	Aims to reduce gradient overfitting	[2], [20], [24]	Independent of target attack characteristics	Can reduce attack accuracy, limited transferability
Target Modification	Focuses on common features in different models	[13], [21], [4], [10]	Exploits model similarities	Relies on model feature commonalities, difficult to implement
Ensemble Approach	Integrates multiple models for attacks	[8], [17]	Enhances transferability and robustness	More computationally intensive, requires multiple model integration

III. SYSTEMIZATION OF KNOWLEDGE

In this section, we provide a comprehensive summarisation to a variety of existing transferable attack methods, each enhancing the transferability of adversarial attacks from different perspectives. The first category is for baseline methods, which, while not specifically optimised for attack transferability, utilize classic white-box attack methods to assess the effectiveness of other transferable methods. As depicted in Table I, we categorise the transferable attacks into five types: Generative Architecture, Semantic Similarity, Gradient Editing, Target Modification, and Ensemble Approach.

A. Baseline approaches

To study transferable adversarial attacks, the selection of a suitable baseline method is essential for benchmarking attack techniques. In this work, we select the Iterative Fast Gradient Sign Method (I-FGSM) as the baseline [2], [16]. As an enhancement of FGSM, I-FGSM applies iterative refinements to generate more effective adversarial samples. Its lack of specific optimisation for transferability provides unified context for assessing other methods. Superior performance over I-FGSM in transferability is indicative of enhanced attack efficacy.

The principle of I-FGSM involves iterative applications of the Fast Gradient Sign Method. Starting with an input x_0, each iteration computes the loss function gradient L relative to the current sample x_i, aiming to maximize loss and induce misclassification. Adjustments are based on the gradient sign, following $x_{i+1} = x_i + \epsilon \cdot \text{sign}(\nabla_x L(\theta, x_i, y))$, where ϵ controls perturbation magnitude, θ denotes model parameters, and y the true label. The update size is consistent across pixels, guided by the gradient direction. The process concludes after a predetermined number of iterations or upon achieving misclassification.

B. Generative Architecture

This category of methods employs Generative Adversarial Networks (GANs) to produce adversarial samples. The core idea of these methods lies in using generative models to mimic the decision boundaries of the target attack models, thereby generating efficient transferable adversarial samples. The advantage of such methods is that once the Generator is trained, adversarial samples can be quickly generated without further querying the model. Examples of this type of attack include AdvGAN [15] and GE-AdvGAN [24].

1) AdvGAN [15]: The main principle of AdvGAN is based on Generative Adversarial Networks (GANs), which include a generator and a discriminator. The generator creates slight perturbations and adds them to the original input data, generating counterfeit samples. The discriminator learns to distinguish between fake and real samples, which are then assessed by the target neural network to evaluate the classification effect of the perturbed samples. The adversarial samples generated by AdvGAN aim to deceive the target network into making incorrect classifications while remaining imperceptible to human observers. This method effectively combines the generative capabilities of GANs with the requirements of adversarial attacks, functioning efficiently in both semi-white-box and black-box attack scenarios.

The overall objective of AdvGAN is to find a balance between generating adversarial samples and deceiving the model. Thus, the total loss is the sum of the discriminator loss and the generator's impact on the target model:

$$\min_G \max_D \mathbb{E}_{x,y\sim\text{data}}[\log D(x)+\log(1-D(G(x)))+\lambda L(f(G(x)),y)]$$

, where λ is a weight coefficient used to balance the two objectives. Through this method, AdvGAN can generate adversarial samples that are as similar to real samples as possible but can mislead the target model.

2) GE-AdvGAN [24]: GE-AdvGAN, compared to AdvGAN, has been optimized in terms of transferability and has also improved the efficiency of the algorithm. The core idea is the optimization of the gradient update method during the generator training process. GE-AdvGAN introduces a novel Gradient Editing (GE) mechanism, utilizing frequency domain exploration to determine the direction of gradient editing. This method enables GE-AdvGAN to generate highly transferable adversarial samples while significantly reducing the execution time to generate these samples.

Specifically, in AdvGAN, the Generator's loss is divided into three components: L_{adv}, L_{GAN}, and L_{hinge}. These respectively represent the loss functions for the attack, generation, and control of perturbations. The portion controlling the attack can be decomposed as $\nabla\theta_G L_{adv} = \frac{\partial L_{adv}}{\partial(x+G(x))} \cdot \frac{\partial(x+G(x))}{\partial G(x)} \cdot \frac{\partial G(x)}{\partial \theta_G}$. In GE-AdvGAN, the term $\frac{\partial(x+G(x))}{\partial G(x)}$ is replaced with $-\text{sign}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{\partial L(x_{f_i},y)}{\partial x_{f_i}}\right)$, where x_{f_i} are samples generated using frequency domain exploration.

C. Semantic Similarity

The core concept of this category of methods is to find a sample and construct samples that are semantically related to it, and simultaneously attack these semantically related samples, thereby expanding the transferability of adversarial attacks. Such attack methods are represented by Diverse Input Fast Gradient Sign Method (DI-FGSM) [16], Scale-Invariant Nesterov Iterative Fast Gradient Sign Method (SI-NI-FGSM) [7], Spectrum Simulation Attack (SSA) [9], Centralized Perturbation Attack (CPA) [14], Feature Disruptive Universal Adversarial Attack (FDUAA) [11], and Structure Invariant Attack (SIA) [12].

1) DI-FGSM [16]: The core principle of the DI-FGSM is to introduce input diversity in the process of generating adversarial samples to find Semantic Similarity. This is achieved by applying random transformations (such as resizing and padding) to the input image in each iteration. These variations create diverse input patterns, helping to prevent overfitting to specific network parameters, thereby enhancing the effectiveness of the generated adversarial samples against different models.

Assume the original input image is x, and the adversarial sample is initialized as $x'_0 = x$. For each iteration i, a random transformation τ is applied to the current adversarial sample x'_i, resulting in the

transformed sample $\widetilde{x}_i' = \tau(x_i')$. Then, the gradient of the loss function $L(f(\widetilde{x}_i'), y)$ with respect to $\widetilde{x}_i'$ is computed, where y is the target label. The adversarial sample is updated using this gradient:

$$x_{i+1}' = x_i' + \epsilon \cdot \text{sign}(\nabla_{x_i'} L(f(\widetilde{x}_i'), y))$$

Here, ϵ is the step size, and the sign function returns the sign of the gradient. By repeating this process, DI-FGSM increases the transferability of the adversarial samples, making them more likely to be effective on unknown models.

2) SI-NI-FGSM [7]: The SI-NI-FGSM is an adversarial attack algorithm that integrates Scale Invariance (SIM) and the Nesterov Iterative Method (NIM). This method enhances the effectiveness and transferability of adversarial samples by introducing NIM and SIM on top of the Fast Gradient Sign Method (FGSM). SI-NI-FGSM first utilizes NIM to predict future changes in the gradient for more precise updates of the adversarial samples. It then maintains scale invariance by adjusting the scale of the input image, thus improving the transferability of the attack across different models.

Specifically, SI-NI-FGSM initially pre-updates the input sample using the Nesterov method, calculated with the formula $x' = x + \alpha \cdot v$, where x is the current sample, v is the accumulated gradient, and α is the pre-update step length. It then calculates the gradient at the pre-update point $g = \nabla_x L(\theta, x', y)$ and updates the momentum $v = \mu \cdot v + g$. Finally, the sample is updated using $x = x + \epsilon \cdot \text{sign}(v)$. During the generation of adversarial samples, the scale of the input image is adjusted to ensure that the generated perturbation maintains the same effect on images of different scales.

3) SSA [9]: The core principle of SSA is to simulate diverse models in the frequency domain, thereby enhancing the transferability of samples. The specific operation includes using DCT and inverse DCT to transform the spectral signature of the input image, generating diverse spectral saliency maps, which indicate the diversity of substitute models. The approach further includes randomly masking features in the frequency domain to identify and exploit similar semantics, thereby accomplishing transferability in the attack.

Specifically, SSA first uses DCT to transform the input image from the spatial domain to the frequency domain. This process can be mathematically represented as $\mathcal{D}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x}\boldsymbol{A}^{\mathrm{T}}$, where $\boldsymbol{A}$ is an orthogonal matrix. Subsequently, SSA introduces a Spectrum Saliency Map, defined as the response of the spectrum of the input image to the gradient of the model's loss function, expressed as $\boldsymbol{S}_\phi = \frac{\partial L(\mathcal{D}_\mathcal{I}(\mathcal{D}(\boldsymbol{x}), y; \phi)}{\partial \mathcal{D}(\boldsymbol{x})}$. Here $\mathcal{D}_\mathcal{I}$ is the inverse DCT transform, used to convert frequency domain data back to the spatial domain. Finally, SSA employs a random spectral transformation $\mathcal{T}(\cdot)$, which can be expressed as $\mathcal{T}(\boldsymbol{x}) = \mathcal{D}_\mathcal{I}(\mathcal{D}(\boldsymbol{x}) + \mathcal{D}(\boldsymbol{\xi}) \odot \boldsymbol{M})$, where $\odot$ denotes the Hadamard product, and ξ and $\boldsymbol{M}$ are variables randomly sampled from Gaussian and uniform distributions, respectively. This transformation produces diverse spectral saliency maps, thereby simulating different substitute models and enhancing the transferability of adversarial samples.

4) CPA [14]: The principle of CPA [14] is to enhance the transferability of adversarial attacks through precise perturbation optimization in the frequency domain on DNNs. This method first employs DCT to decompose data into the frequency domain, thereby facilitating the exploration of similar semantics. Then, it reduces unnecessary perturbations by quantizing each Y/Cb/Cr channel and focuses the optimization on the main frequency coefficients that influence model predictions. Finally, the differential quantization matrix is optimized through backpropagation, ensuring that perturbations are concentrated in the dominant frequency areas. The key to this method lies in effectively centralizing and optimizing perturbations, thereby improving the transferability of adversarial samples and their ability to bypass defense mechanisms.

5) FDUAA [11]: The core principle of the FDUAA is to generate universally transferable adversarial perturbations (UAPs) by disrupting features that are not dependent on specific model architectures, such as edges or simple textures. Specifically, this method weakens important channel features while enhancing less significant ones, as determined by a specific strategy, through a target function. Additionally, the method iteratively updates UAPs using the average gradient of small-batch inputs to capture local information. It also introduces a momentum term to accumulate gradient information from iterative steps, sensing the global information of the entire training set.

6) SIA [12]: The principle of the SIA is based on applying a series of random transformations to an image, aiming to create diverse adversarial samples with structural characteristics. The SIA method processes the image in blocks, applying random image transformations such as rotation and scaling to each block, thereby increasing the diversity of the samples and finding similar semantics. This method maintains the basic structure of the original image while generating challenging adversarial samples capable of effectively deceiving deep neural networks. The key to SIA lies in its ability to enhance the transferability of the samples by introducing transformations, while simultaneously maintaining the structural integrity of the image.

7) FSPS [25]: The FSPS method is a novel algorithm designed to enhance the transferability of adversarial attacks in machine learning. This approach is centered around two fundamental concepts: identifying stationary points on a loss curve and executing frequency-based searches from these identified points. The process is initiated by pinpointing stationary points on the loss curve, defined as locations where the derivative of the loss function is zero. These identified points serve as the starting points for the attack. Subsequently, FSPS applies a frequency-based search methodology to scrutinize the most effective adversarial directions in the vicinity of these stationary points.

D. Gradient Editing

This category of methods focuses on modifying or optimizing gradient information to generate adversarial samples. These techniques often rely on a deep understanding and manipulation of gradients in surrogate models, to make the generated samples effective on target models. Representative methods include Momentum Iterative Fast Gradient Sign Method (MI-FGSM) [2], Token Gradient Regularization (TGR) [20], Frequency-based Stationary Point Search (FSPS) [25], and the previously mentioned GE-AdvGAN [24]. Since GE-AdvGAN has already been discussed earlier, it will not be elaborated upon in this section.

1) MI-FGSM [2]: The MI-FGSM integrates a momentum term in its iterative process to stabilize the update direction and escape from local maxima, thereby generating more transferable adversarial samples. In each iteration, it accumulates a velocity vector in the direction of the loss function gradient, aiding in optimizing stability and avoiding suboptimal local maxima.

In MI-FGSM, an adversarial perturbation rate α is set, proportional to the total perturbation limit ϵ and the number of iterations T. The method starts with the original input x and initializes a zero vector g as the starting value for momentum. In each iteration, it first calculates the gradient of the loss function $\nabla_x L(x_t, y)$ for the current adversarial sample x_t, then combines this gradient with the previous momentum g_t, weighted by the momentum factor μ, to adjust the direction of the next update. The momentum is updated using $g_{t+1} = \mu \cdot g_t + \frac{\nabla_x L(x_t, y)}{\|\nabla_x L(x_t, y)\|_1}$. The role of momentum is to maintain directionality throughout the optimization process and effectively circumvent falling into local optima. Finally, the new adversarial sample x_{t+1} is iteratively generated using $x_{t+1} = x_t + \alpha \cdot \text{sign}(g_{t+1})$.

2) TGR [20]: The TGR is an adversarial attack method specifically designed for Vision Transformers (ViTs). It enhances the attack effectiveness by reducing gradient variance during the training process. This method leverages the internal structural features of ViTs, diminishing the disparity in gradients among tokens, and thereby scaling the model's sensitivity to specific adversarial samples. Consequently, the adversarial samples generated are more likely to mislead different ViT models when transferred, inducing incorrect judgments. Notably, TGR demonstrates high efficacy and transferability in adversarial settings against various ViT and CNN models.

E. Target Modification

This category of methods exploits the characteristic of similarity among different models, such as attribution (explainability) similarity, by directly attacking these similar features to achieve transferable goals. Instead of directly using the model's cross-entropy, these methods often target the model's intermediate layers or attributions, such as Feature Importance-Aware Attack (FIA) [13], Neuron Attribution-based Attack (NAA) [21], Double Adversarial Neuron Attribution Attack (DANAA) [4], and Momentum Integrated Gradients (MIG) [10].

1) FIA [13]: The FIA achieves attack transferability by targeting key object-aware functions that significantly impact model decisions. Unlike traditional methods that indiscriminately distort features, leading to overfitting and limited transferability, FIA introduces an aggregated gradient approach. This method averages the gradients of a batch of randomly transformed versions of the image, emphasizing features related to the object and deemphasizing model-specific features. Such gradient information guides the generation of adversarial examples, aimed at disrupting key features, thereby enhancing transferability across different models.

2) NAA [21]: The NAA method first comprehensively attributes the model output to each neuron in the intermediate layer, then significantly reduces the computational cost through an approximation scheme. This scheme is based on two main assumptions: first, the network's feature extraction layers and decision layers are independent in most traditional DNN models; second, that the gradient sequences of these two parts have zero covariance. As a result, NAA can make a faster and relatively accurate estimation of the importance of neurons. By weighting the attribution results of the neurons, it attacks the feature layer, thus generating transferable adversarial examples.

Specifically, NAA first uses the formula $A_{yj} = \sum(x_i - x_i') \int_0^1 \frac{\partial F}{\partial y_j}(y(x_\alpha)) \frac{\partial y_j}{\partial x_i}(x_\alpha) d\alpha$ to calculate neuron attribution. This formula measures the importance of neuron y_j by considering each input feature x_i's effect on neuron y_j and neuron y_j's contribution to the final output F. Then, using the simplified computational assumptions, the attribution formula becomes $A_{yj} \approx \Delta y_j \cdot IA(y_j)$. Here, $IA(y_j)$ is the integrated attention, and this approximation allows for a rapid assessment of neuron importance. Finally, the target of perturbation generation is to minimize the weighted attribution $WA_y = \sum_{A_{y_j} \geq 0} f_p(A_{y_j}) - \gamma \cdot \sum_{A_{y_j} \leq 0} f_n(-A_{y_j})$. This process adjusts the input image to reduce the model output's reliance on positive features while enhancing the impact of negative features, thereby improving the performance of transferable adversarial samples.

3) DANAA [4]: The DANAA method (Double Adversarial Neuron Attribution Attack) is an attack technique based on double adversarial neuron attribution. Its core principle lies in updating perturbations via a non-linear path, thereby more accurately assessing the importance of intermediate-layer neurons. The DANAA method attributes the model output to intermediate layer neurons, measuring the weight of each neuron and retaining features more crucial for transferability. This approach, by improving attribution results, enhances the transferability of adversarial attacks.

4) MIG [10]: MIG utilizes integrated gradient attributions to generate adversarial perturbations. Compared to traditional gradients, integrated gradients exhibit higher similarity across different models. MIG also incorporates a momentum strategy, optimizing the perturbation updates by accumulating integrated gradients from previous iterations, thus enhancing the attack success rate and transferability.

Specifically, MIG starts by generating an initial zero perturbation $\delta_0 = 0$. In each iteration, it calculates the gradient of the loss function of the current input image relative to the model $\nabla_x L(f(x + \delta_t), y)$, and then combines this gradient with the momentum accumulated from previous iterations m_t. The momentum update formula is $m_{t+1} = \mu \cdot m_t + \frac{\nabla_x L(f(x+\delta_t),y)}{\|\nabla_x L(f(x+\delta_t),y)\|_1}$, where μ is the momentum factor. The current perturbation is then updated using the accumulated momentum m_{t+1}, following the formula $\delta_{t+1} = \delta_t + \alpha \cdot \text{sign}(m_{t+1})$, where α is the step size. By iteratively repeating this process, MIG can gradually construct more transferable adversarial perturbations.

F. Ensemble Approach

This category of methods employs an approach where the attack process combines multiple models, using queries from multiple models to enhance transferability. However, these methods have certain limitations in real-world scenarios, as it is challenging to obtain multiple surrogate models in practical applications. Methods that fit this category include Model ensemble attacks [8] and Stochastic Variance Reduced Ensemble Attack (SVRE) [17].

1) Model ensemble attacks: Liu et al. [8] proposed generating more transferable adversarial samples by ensembling multiple models. The core idea of this method is to optimize an ensemble of white-box models to generate adversarial samples capable of deceiving other black-box models. Specifically, given k white-box models with softmax outputs J_1 to J_k, an original image x, and its true label y, the ensemble method solves the following optimization problem:

$$\text{argmin}_x - \log\left(\sum_{i=1}^k \alpha_i J_i(x)\right) \cdot 1_y + \lambda d(x, x')$$

Here, y is the target label specified by the attacker, $\alpha_i J_i(x)$ represents the ensemble model, and α_i are the ensemble weights (satisfying $\sum_{i=1}^k \alpha = 1$). The goal of this optimization is to generate adversarial images that maintain their adversarial nature against an additional black-box model J_{k+1}, retaining transferability for different models.

2) SVRE [17]: The SVRE operates on the principle of reducing gradient variance in model ensemble attacks to improve the transferability of adversarial samples. In traditional model ensemble attacks, attackers simply merge outputs from multiple models, but this approach neglects differences in gradient variance between models, possibly leading to local optima. SVRE reduces the variance through a two-level loop approach: the outer loop computes the average gradient of all models and passes the current sample to the inner loop; the inner loop performs multiple iterative updates, calculating the current gradient on a randomly selected model in each iteration, and adjusting it according to the gradient deviation in the outer loop. This method results in more accurate gradient updates in the outer loop, avoiding the issue of overfitting in ensemble model and enhancing the transferability of adversarial samples to unknown models.

IV. Description of TAA-Bench

A. Algorithms Implementation

In TAA-Bench, we consider 10 different types of adversarial attack methods as the current solution: I-FGSM [5], DI-FGSM [16], MI-FGSM [2], SI-NI-FGSM [7], NAA [21], DANAA [4], SSA [9], MIG [10], AdvGAN [15], and GE-AdvGAN [24]. We select the methods as either classical or state-of-the-art TAA approaches.

Classical methods serve as baselines to measure the advancements of other newly improved algorithms. Moreover, these methods are

chosen for their practical popularity and reproducibility in related studies, since some algorithms may involve a large number of hyperparameters leading to uncertainty in results and difficulties in implementation. In this case, our benchmark does not include such methods. The goal of TAA-Bench is to make the usage of TAA methods as simple and practical as possible to facilitate the in-depth analysis. We reflect the limitation of TAA-Bench by continuously including latest research results in our benchmark.

B. Codebase of TAA-Bench

We have constructed an extensible and modular codebase as the foundation for TAA-Bench, including three modules: configuration, attack, and network model modules.

The configuration module comprises a YAML file for defining experimental parameters. This setup facilitates adaptable, reproducible experiments by clearly outlining variables such as network specifications and algorithm hyperparameters. Ensuring consistency and ease of modification, this module aligns with the imperative of reproducibility in scientific research.

The attack module, employing a modular architecture, encapsulates all the adversarial attack methods. The module aids future researchers for code review or extending new methods. Thus, the module provides a universal, dynamic toolkit to simulate and analyse the performance of transferable adversarial attack methods.

The network model module incorporates ten classic models: Inception-v3, Inception-v4, ResNet-50, ResNet-101, ResNet-152, Inception-ResNet-v2, Inception-v3-adv, Inception-v3-ens3, Inception-v3-ens4, Inception-ResNet-v2-ens-adv in PyTorch. These models ensure that all attack methods can be thoroughly tested under the same structures, ensuring fairness in comparative experiments. Additionally, specific model structures can be added for assessment and testing.

V. Conclusion

In summary, this paper provides an extensive review and benchmarking of the state-of-the-art techniques in the transferability of adversarial attacks, offering significant insights into the field of machine learning security. We have conducted a thorough analysis and categorization of a variety of methods. Our benchmarking efforts of TAA-Bench cover ten different adversarial attack methods, providing a comprehensive assessment of their effectiveness across various model architectures. In future work, we will expand our benchmark and incorporate data analysis methodologies, such as interpretability analysis, to conduct an exhaustive evaluation of all methods.

References

[1] A. Chakraborty, M. Alam, V. Dey, A. Chattopadhyay, and D. Mukhopadhyay, "Adversarial attacks and defences: A survey," *arXiv preprint arXiv:1810.00069*, 2018.

[2] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, "Boosting adversarial attacks with momentum," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9185–9193.

[3] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[4] Z. Jin, Z. Zhu, X. Wang, J. Zhang, J. Shen, and H. Chen, "Danaa: Towards transferable attacks with double adversarial neuron attribution," in *International Conference on Advanced Data Mining and Applications*. Springer, 2023, pp. 456–470.

[5] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Artificial intelligence safety and security*. Chapman and Hall/CRC, 2018, pp. 99–112.

[6] Y. Li, M. Cheng, C.-J. Hsieh, and T. C. Lee, "A review of adversarial attack and defense for classification methods," *The American Statistician*, vol. 76, no. 4, pp. 329–345, 2022.

[7] J. Lin, C. Song, K. He, L. Wang, and J. E. Hopcroft, "Nesterov accelerated gradient and scale invariance for adversarial attacks," *arXiv preprint arXiv:1908.06281*, 2019.

[8] Y. Liu, X. Chen, C. Liu, and D. Song, "Delving into transferable adversarial examples and black-box attacks," *arXiv preprint arXiv:1611.02770*, 2016.

[9] Y. Long, Q. Zhang, B. Zeng, L. Gao, X. Liu, J. Zhang, and J. Song, "Frequency domain model augmentation for adversarial attack," in *European Conference on Computer Vision*. Springer, 2022, pp. 549–566.

[10] W. Ma, Y. Li, X. Jia, and W. Xu, "Transferable adversarial attack for both vision transformers and convolutional networks via momentum integrated gradients," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4630–4639.

[11] D. Wang, W. Yao, T. Jiang, and X. Chen, "Improving transferability of universal adversarial perturbation with feature disruption," *IEEE Transactions on Image Processing*, 2023.

[12] X. Wang, Z. Zhang, and J. Zhang, "Structure invariant transformation for better adversarial transferability," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4607–4619.

[13] Z. Wang, H. Guo, Z. Zhang, W. Liu, Z. Qin, and K. Ren, "Feature importance-aware transferable adversarial attacks," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 7639–7648.

[14] S. Wu, Y.-a. Tan, Y. Wang, R. Ma, W. Ma, and Y. Li, "Towards transferable adversarial attacks with centralized perturbation," *arXiv preprint arXiv:2312.06199*, 2023.

[15] C. Xiao, B. Li, J.-Y. Zhu, W. He, M. Liu, and D. Song, "Generating adversarial examples with adversarial networks," *arXiv preprint arXiv:1801.02610*, 2018.

[16] C. Xie, Z. Zhang, Y. Zhou, S. Bai, J. Wang, Z. Ren, and A. L. Yuille, "Improving transferability of adversarial examples with input diversity," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 2730–2739.

[17] Y. Xiong, J. Lin, M. Zhang, J. E. Hopcroft, and K. He, "Stochastic variance reduced ensemble adversarial attack for boosting the adversarial transferability," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 14 983–14 992.

[18] H. Xu, Y. Ma, H.-C. Liu, D. Deb, H. Liu, J.-L. Tang, and A. K. Jain, "Adversarial attacks and defenses in images, graphs and text: A review," *International Journal of Automation and Computing*, vol. 17, pp. 151–178, 2020.

[19] C. Zhang, P. Benz, A. Karjauv, J. W. Cho, K. Zhang, and I. S. Kweon, "Investigating top-k white-box and transferable black-box attack," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 15 085–15 094.

[20] J. Zhang, Y. Huang, W. Wu, and M. R. Lyu, "Transferable adversarial attacks on vision transformers with token gradient regularization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 16 415–16 424.

[21] J. Zhang, W. Wu, J.-t. Huang, Y. Huang, W. Wang, Y. Su, and M. R. Lyu, "Improving adversarial transferability via neuron attribution-based attacks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 14 993–15 002.

[22] Z. Zhao, Z. Liu, and M. Larson, "On success and simplicity: A second look at transferable targeted attacks," *Advances in Neural Information Processing Systems*, vol. 34, pp. 6115–6128, 2021.

[23] S. Zhou, C. Liu, D. Ye, T. Zhu, W. Zhou, and P. S. Yu, "Adversarial attacks and defenses in deep learning: From a perspective of cybersecurity," *ACM Computing Surveys*, vol. 55, no. 8, pp. 1–39, 2022.

[24] Z. Zhu, H. Chen, X. Wang, J. Zhang, Z. Jin, and K.-K. R. Choo, "Ge-advgan: Improving the transferability of adversarial samples by gradient editing-based adversarial generative model," *arXiv preprint arXiv:2401.06031*, 2024.

[25] Z. Zhu, H. Chen, J. Zhang, X. Wang, Z. Jin, Q. Lu, J. Shen, and K.-K. R. Choo, "Improving adversarial transferability via frequency-based stationary point search," in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 2023, pp. 3626–3635.