# A SURVEY OF ATTACK AND DEFENSE METHODOLOGY ON ADVERSARIAL MACHINE LEARNING

**Ajay Kesarwani**[*1]

[1]University of Passau, Germany

*Correspondance : kesarw01@ads.uni-passau.de

**ABSTRACT**

In recent years, machine learning technologies have been used in almost every aspect of the area including industry and academic institutions. However, it has been observed that machine learning models are highly vulnerable to adversarial examples where attackers manipulate the model through subtle changes in data which can cause it to function abruptly. Hence, both its performance and success ratio of predicting the expected outcomes are completely compromised. In this survey paper, we will expose several attacking methodologies used by the adversary, as well as several defense algorithms that have been proposed by scholars to overcome these attacks and build a high performing robust model. Through the lens of comparative study, it is essential to know each one of the popular adversarial attacks in order to decide which defense algorithm should we use and when?

**Keywords:** adversarial machine learning, neural network, adversarial example, adversarial attack, adversarial defense.

## 1 INTRODUCTION

Adversarial machine learning is a machine learning technique that attempts to exploit models by taking advantage of obtainable model information and using it to create malicious attacks. The most common reason is to cause a malfunction in a machine learning model [1]. The machine learning model is built using training data and then predicts output based on unseen input data. Attackers learn the nature of the model to manipulate both the data and the model by malicious inputs.

Machine Learning Algorithms have usage in most sectors including industry and academic institutions. As the rapid development of internet technology is rising in every sector so is the use of machine learning models in order to predict the correct outcomes based on given input data. However, it has been seen from the last few decades that attackers try to change the behaviour of the machine learning model through maliciously tampering with the input data or retraining the model with false data or some other ways so that the target model produces the output according to their desired values. Since the machine learning model is also widely used in a few safety-critical domains like health care, autonomous driving, finance, and national security which could cause the risk to the life and health of the common people directly or indirectly. So it is essential to achieve better performing machine learning models that should be robust enough against attackers and avoid such possible attacks. Therefore, in recent years several researchers from academia and tech companies have conducted experiments with different new approaches to identify such attacks and overcome them.

Adversarial attacks are more frequently occur to create a robust and high performing machine learning model that defends itself against adversarial attacks, we need to first understand all possible weak spots that the attackers use, to exploit the model and gets success in modifying the behaviour of the model. The well-known way is that the attacker manipulates the model by feeding false data. Therefore, the first step is to understand the kind of dataset that is being used to train the model then the attacker's ability to access the model's architecture and parameter values.

We have noted from the recent studies that data are mostly related to image classification, text and audio. There are mainly 3 different types of datasets involved to create an accurate machine learning model. The training datasets are used to train the model in the training phase, the validation dataset to improve the quality of the model in the evaluation phase and finally, the unseen test dataset to check the success rate of the model in the deployment phase.

If an adversary has access to the model architecture and parameter values, it is possible to replicate the targeted system on its own machine. This kind of scenario is known as the white box. Whereas in the black box scenario,

the attackers do not have any knowledge about the machine learning algorithm and the parameter values. Therefore, attackers try to understand the behaviour of the model based on its predictions or must rely on guesswork. A full thorough and comparative study between white box and black box attacks using examples will be done in the following sections.

There are several different kinds of methods generally used by attackers to manipulate the model such as data poisoning, evasion, and model stealing are few among them. The data poisoning attacks occur in the training phase, where the attackers first create a manipulated small portion of the dataset and, mix them into the training dataset. Once the model is trained with these modified mixed training datasets, the attacker uses previously mixed manipulated data to trigger specific behaviour in the model during inference time. The evasion attack is commonly seen in email spamming where attackers learn the model's output prediction on input data, and they attack the model by adding some good keywords into the input data. Finally, model stealing is a more sensible attack where attackers reconstruct the model or try to obtain confidential training data.

In this paper, we focus on achieving better performing machine learning models that are robust against adversarial attacks. First, we will start by understanding adversarial machine learning attacks. Then explore different defense algorithms for protecting against adversarial examples used by researchers where each one of them has some strengths and weaknesses depending on some factors. For instance, Szegedy et al. first used [2] Limited memory Broyden Fletcher Goldfarb-Shanno (L-BFGS) method which is a nonlinear gradient-based optimization algorithm. They did the smallest possible perturbation in the input data and found a high success ratio but fails to provide low computational complexity. Goodfellow et al. [3] proposed a method named the FastGradient Sign method (FGSM) which is one of the fastest and cheapest ways to implement but have a low success rate. The methods like the Basic Iterative method (BIM) [4], Projected gradient descent (PGD) [5], Adversarial Transformation Networks (ATNs) [6], Jacobian-based Saliency Map Attack (JSMA) [7], Carlini Wagner (C&W) [8] are explained in the upcoming sections.

This survey paper is organized as follows: Section 2 explore the related work done by researchers in recent years. Section 3 will give some background on this topic using some illustrations. Section 4 will review the attack methodology that has been popularly used by the adversaries to generate adversarial examples. Section 5 will describe the defense mechanism introduced by researchers to overcome adversarial examples. Finally, Section 6 will discuss some of the interesting results of these attack and defense methods and will conclude by giving a thought for future work to create high performing machine learning model that must be robust enough so that it cannot be manipulated by any adversaries.

## 2 RELATED WORK

The adversarial attack has been comprehensively studied for understanding the robustness of machine learning models. Our surveys focus only on the related papers and explore the attacker's and defender's ways of thinking. For instance, what kind of approaches the attackers in general use and how to countermeasure from the defender's point of view. Our main contribution is to comparatively expose the relevant work done in this space in recent years and to give a thought and new direction while protecting the model against adversarial attacks. We focus on related work on both white box and black-box adversarial attacks in the machine learning model.

**Adversarial Attacks** The first adversarial example was introduced by Szegedy et al. [2] using the L-BFGS method. From then many researchers came together and have proposed several different methods and techniques to efficiently get a better result. For instance, Goodfellow [3] proposed Fast Gradient Sign Method (FGSM), Kurakin et al. [4] introduced the Basic Iterative Method, Madry et al. [5] introduced Projected Gradient Descent and many more. While doing the experiments researchers also found different new things. For instance, Dalvi et al. [9] found that using evasions attack, spammers can modify the spam filter by using some good keywords. Battista et al. [10] used a first-time gradient-based approach against evasion attacks in test time [10]. Szegedy et al. [2] found that a small perturbation of original input images would give completely different results [2]. While Goodfellow et al. [3] believe that applying small perturbation in input could cause model generating outcomes with a high confidence value. Nicholas Carlini and David Wagner [8] introduced a method to determine perturbations with minimal $l_p$ norms by simultaneously minimizing the perturbations. These attacks rely on the transferability of adversarial examples in a black-box setting. It has been demonstrated by Liu et al. [11] that these examples have little transferability to attacks. In contrast, Cheng et al. [12] presented a score-based attack method based on the zeroth-order attack with gradient estimation. Tu et al. [13] further improved this method. Zhou et al. [14] introduced data free substitute training method where they found competitive performance.

**Adversarial Defenses** In order to increase the robustness of models, several defense methods have been proposed. In adversarial training [2, 3, 4], each model directly trains on adversarial examples. Another method aims to modify the adversarial examples themselves such as local linearity regularization [15]. While some attacks can be prevented by the above defenses, they can still be exploited. Additionally, researchers are interested in detecting adversarial examples. Some of them detect whether or not the examples are adversarial [16, 11, 17, 18, 19, 20]. Some certified training defense methods have also been proposed [21, 22, 23] which provided a significant improvement in the defense. Another defense method using the Generative Adversarial framework has also been proposed by researchers [24, 25, 26, 27, 28, 29].

## 3 BACKGROUND

In this section, we will discuss the adversarial attacks and explain them with the help of images. An adversary provides adversarial examples to machine learning models that are meant to cause the machine to make a mistake. Szegedy et al. [2] first trained the neural networks with the mixture of adversarial examples and clean examples so that models can learn to classify correctly between them. In their experiment, they were able to generate the adversarial examples for different datasets like MNIST, QuocNet and AlexNet, which are visually hard to distinguish. Goodfellow [3] further enhanced his method and get better performance. For instance, figure 1 shows how a panda is detected as a gibbon by the model with just .007 magnitude of the smallest bit of 8-bit image noise was added to the original image. Here they found that the confidence score for gibbon the image is 99.3% but for panda is only 57.7%.

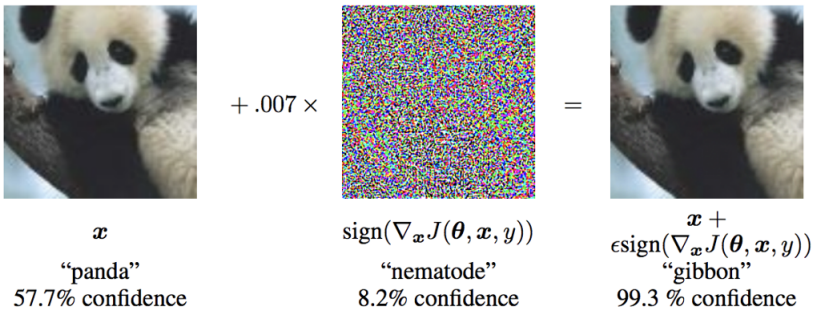


Figure 1: A demonstration of fast adversarial example generation applied to GoogLeNet(Szegedy et al. [2]) on ImageNet. By adding an imperceptibly small vector whose elements are equal to the sign of the element of the gradient of the cost function with respect to the input, we can change GoogLeNet's classification of the image. Here our $\epsilon$ of .007 corresponds to the magnitude of the smallest bit of an 8-bit image encoding after GoogLeNet's conversion to real numbers. [3]

In another experiment done by Eykholt et al. [30] where they proposed an algorithm where they show that adversary can effectively modify the physical objects. For instance, the attacks cause a classifier to interpret a subtly-modified physical Stop sign as a Speed Limit 45 sign. They only added a set of black and white stickers to the original that an adversary can attach to a physical road sign (Stop sign) as shown in Figure 2. The perturbations were designed to look like graffiti. In the real world, we often see road signs with graffiti or color alterations. As adversarial perturbations, these patterns could harm autonomous driving systems without evoking suspicion from humans.


Figure 2: The left image shows real graffiti on a Stop sign, something that most humans would not think is suspicious. The right image shows a physical perturbation applied to a Stop sign. Perturbations is designed to mimic graffiti, and thus "hide in the human psyche" [30]

# 4 METHODS GENERATING ADVERSARIAL EXAMPLES

## 4.1 TERMINOLOGY AND NOTATION

In this paper we use the following definitions and notations used regarding adversarial examples:

- x, the original input data of the model which can either training or test datasets
- $x'$, the adversarial perturbed data
- y, the original target label associated with x
- $\epsilon$, the size of the small adversarial perturbation.
- $J(\theta, x, y))$, the cost function used to train the model
- c is a constant
- $\eta$ is the perturbation
- $\theta$ is the parameters of the target model

## 4.2 ATTACK METHODS

### *4.2.1 Limited memory Broyden Fletcher Goldfarb-Shanno (L-BFGS)*

Szegedy et al. [2] first introduced the Limited memory Broyden Fletcher Goldfarb-Shanno (L-BFGS) method that generates adversarial examples using a white-box attack. They trained the target model with a mixture of adversarial examples and clean examples so that model can learn to classify correctly between them. This method is based on a nonlinear gradient. They found a high success ratio but fails to provide low computational complexity. It can be expressed as:

$$min_{x'} c||\eta|| + J_\theta(x', l') \qquad s.t. \quad x' \in [0, 1] \tag{1}$$

This method finds different $x'$ images for the given original input image x that is almost similar to the original image. Szegedy et al. [2] attempt to minimize this difficult problem 1. An adversarial example with minimum distance is generated based on constant $c \geq 0$

### *4.2.2 Fast Fradient Sign Method (FGSM)*

Goodfellow et al. [3] introduced the Fast Gradient Sign Method (FGSM) to reduce the computation complexity of L-BFGS in order to generate adversarial examples. This is the simplest attacking method where the attackers fool the model by simply adding errors of the network with respect to the given input also called the gradient. This method is a single-step attack that adds perturbation along the direction of the gradient where the gradient gives an idea about how much amount of error should be increased which allow faster computation and better memory efficiency. This property helps the attacker to not perturb the input with high value. It can be expressed as:

$$\eta = \epsilon sign(\nabla_x J(\theta, x, y)) \tag{2}$$

Here $\epsilon$ should be small enough so that it can not be detected. This method checks the gradient of the loss function for each pixel and to minimize the loss function it gradually increases or decreases the value.

### *4.2.3 Basic Iterative Method*

Based on FGSM, Kurakin et al. [4] introduced the Basic Iterative Method (BIM), which is similar to FGSM but can be performed multiple times to generate adversarial examples. They used the multiple small steps $\alpha$ against the single step size of small perturbation which ends up better result than the FGSM. The number of iterations depends on the parameter chosen by the adversary. It can be expressed as:

$$x_0{}' = 0 \tag{3}$$

where each iteration can be expressed as

$$x_i{}' = x'_{i-1} - clip_\epsilon(\alpha sign(J(\theta, x'_{i-1}, y))) \tag{4}$$

Here multiple smaller steps $\alpha$ is used against the single step size of $\epsilon$ in the direction of gradient sign. Finally, the result is clipped by $\epsilon$

#### *4.2.4 Projected Gradient Descent (PGD)*

Madry et al. [5] introduced Projected Gradient Descent. It is considered a generalized version of BIM. It is a multi-step variant on the negative loss function which is more a powerful adversary than FGSM. Their experiment result shows that the adversarial examples have better transferability with this method and significantly increased the robustness of neural networks and provide the security guarantee protection from any adversary. However, when trained to large datasets, the computational complexity becomes quite high. It can be expressed as:

$$x'_{i+1} = \prod_{x+S} (x_i + \alpha \ sign(\nabla_x J(\theta, x, y))) \tag{5}$$

where $\prod$ denotes the projection operator, which clips the input at the positions around the predefined perturbation range. $\alpha$ means a gradient step size, and x+S represents the perturbation set. $J(\theta, x, y)))$ is the cross-entropy loss.

#### *4.2.5 Adversarial Transformation Networks (ATNs)*

Baluja et al. [6] introduced Adversarial Transformation Networks (ATNs). Instead of the perturbation generation in adversarial training, it uses a parameterized generator network. The target model does not need gradient computation, so it is faster in computation. ATNs can be performed both black-box or white-box. However, this method is slow and it can be problematic when training large datasets like ImageNet, where it is difficult to construct GANs that cover the entire image. It can be expressed as

$$g_{f,\theta}(x) : x \in X \rightarrow x' \tag{6}$$

where $\theta$ is the parameter vector of g, f is the target network which outputs a probability distribution across class labels, and $x' \sim$ x, but argmax f(x) $\neq$ argmax $f(x')$.

### 4.3 White-Box Attack

In a white-box scenario, the adversary has full access to the model, which entails that the adversary knows what sort of machine learning algorithm is used as well as the values of the model's parameters which makes it easier for the attacker to create adversarial examples. There are a few white-box methods of generating the adversarial examples are mentioned in section 4.2. The Algorithm described in section 4.2 is one of the popular methods used in recent years. Each one of them has its own strengths and weaknesses.

### 4.4 Black-Box Attack

In a Black-Box scenario, the adversary does not know both the machine learning algorithm and the parameters which reflects it to be more practical to real life. In this scenario, attackers have only access to the output of the model which makes them harder to attack. Therefore, they try to mimic the parallel substitute model from scratch after thoroughly observing the outputs of the model or confidence scores. If attackers are not able to create the substitute model then they use the Query Feedback mechanism [31] where they fabricated the input data by perturbing a small amount and querying the model to observe the output. This strategy is known as the transferable attack strategy. A much better architecture can also be used by the attacker than the original architecture for estimating weight. Here, we would be exploring the adversarial example attack based on two main approaches, gradient estimation and substitute model.

#### 4.4.1 *Gradient Estimation*

Lesser the number of queries better will be the attack, as the time taken to start an attack will be minimized. Inspired by Carlini & Wagner (C&W) [8] white-box adversarial attacks Chen et al. [12] introduced the zeroth-order optimization(ZOO) based attacks to directly estimate the gradients of the targeted model to produce black-box adversarial image generation. This method has a comparatively high success rate for adversarial attacks and attains high performance but its computation time is quite high due to a large number of the queries to the target model although they used the attack-space dimension reduction, hierarchical attacks and importance sampling techniques. The main purpose of having a less number of queries is to make the attack smoother and minimize the time taken with the significant effect of an attack.

Tu et al. [13] introduced the Autoencoder-based Zeroth Order Optimization (AutoZOOM) framework for query-efficient black-box attacks. They have two main pillars first is an adaptive random gradient estimation method that balances query counts and distortion, and second, an autoencoder that either learns offline from unlabeled data or perform bilinear resizing for attack acceleration. They found a significantly reduced model queries with a high attack success rate while producing adversarial examples generation.

Chen et al. [32] also introduced another algorithm called HopSkipJumpAttack that uses the binary information at the decision boundary to compute the gradient direction. This algorithm has high performance and also require significantly less number of queries to the target model.

#### 4.4.2 *substitute Model*

There are several works that are based on both the transferability of adversarial examples and the model queries for black-box attacks where the adversary train a parallel model called substitute model to mimic the original model. Papernot et al. [33] first used a substitute model using a synthetic adversarial generated dataset where output labels, assigned by the target deep neural networks through queries to generate adversarial examples.

It is observed that by sending the sequence of queries to the model in neural networks some proprietary information can be exposed which makes the system more vulnerable. Seong et al. [34] introduced a technique that uses a meta-model that predicts the internal information of the model through a sequence of queries.

Zhou et al. [14] demonstrated how to construct substitute models for adversarial black-box attacks with a data-free substitute training method (DaST) with the help of generative adversarial networks (GANs). They found that by training the same set of unrealistic training data, DaST generates competitive performance compared with the baseline models.

### 4.5 Comparative Analysis of Attack Method

Below you will find the most popular attack methods used by the adversaries in white-boxes and black-boxes scenarios, along with some interesting results from their experiments. In addition, this table shows how the proposed method is superior to the previously proposed method.

Table 1: Attack Method

| **Method Name** | **Author** | **Remarks** |
|---|---|---|
| L-BFGS | Szegedy et al. [2] | High success ratio but computation complexity is also high in large datasets |
| FGSM | Goodfellow et al. [35] | Better than L-BFGS for large dataset |
| BIM | Kurakin et al. [4] | Higher Performance than FGSM |
| PGD | Madry et al. [5] | More powerful than FGSM, better transferability and robustness |
| ATNs | Baluja et al. [6] | Gradient free therefore Computation is faster |
| C&W | Carlini & Wagner [8] | Uses white box adversarial attacks |
| ZOO | Chen et al [12] | High performance but Computational time is high |
| AutoZOOM | Tu et al [13] | Better than ZOO |
| HopSkipJumpAttack | Chen et al [32] | High performance |
| Substitute model | Papernot et al. [33] | To generate adversarial examples |
| Meta-model | Seong et al. [34] | Predicts the model by a sequence of queries |
| DaST | Zhou et al. [14] | High performance than a traditional method like above |

## 5 DEFENSE METHODOLOGY

In this section, we will explore all possible ways of defense against adversarial attacks where we will mainly focus on adversarial example detection, adversarial training, preprocessing of data and Generative Adversarial Network (GAN) defence strategy.

### 5.1 Detecting adversarial Examples

In advance, one cannot figure out what type of adversarial attack method will be used by the adversary therefore the best way of defending the machine learning model is to detect the adversarial examples in the initial stage only which would avoid doing further processing. In this approach, the model learns the properties of the original data and understand how it is different from adversarial examples and with this knowledge it filtered out the adversarial examples from the large set of datasets. The detection mechanism heavily relies on the fact adversarial data and original data are fundamentally distinct.

Grosse et al. [16] introduced an approach for detecting the adversarial examples using statistical tests where they modified the target model with an additional output so that it can distinguish between original data and adversarial data. Although their approach successfully detects the adversarial examples when tested against the adversarial data generated from FGSM and Jacobian-based Saliency Map Approach (JSMA) [7] attacks, however, it is not effective against the secondary adversarial attack.

Liu et al. [11] understand this problem and proposed a new method, in which they used steganalysis for adversarial examples detection and further extend the steganalysis features based on the probability of modifications done by adversarial attacks. They found that their method can accurately detect adversarial examples. Their high-dimensional artificial features and Fisher Linear Discriminant based method could not be applied directly to detect secondary adversarial attacks because it is not based on a neural network.

Ma et al. [17] proposed a new approach where they used the local intrinsic dimensionality (LID) to characterize the dimensional properties of adversarial regions to detect the adversarial example detection. Their experimental results showed that the LID of adversarial examples is much higher than the LID of original data. Xu et al. [18] introduced a feature squeezing (FS) detection technique that reduces the search space available to an adversary. They proposed two feature squeezing methods, reducing the color bit depth of each pixel and spatial smoothing. Tian et al. [19] introduced the Sensitivity Inconsistency Detector (SID) method for adversarial detection which has better performance and superior generalization capabilities than local intrinsic dimensionality and features squeezing methods.

Aldahdooh et al. [20] introduced the Selective and Feature-based Adversarial Detection (SFAD) method that uses the uncertainty SelectiveNet method and processes model layers output in order to generate new confidence probabilities. Their experimental results show that their approach has better performance than the state-of-the-art algorithms.

Sutanto et al. [36] introduced a real-time adversarial detection method called Deep Image Prior(DIP) to detect adversarial examples, in which they used blurring network as the initial condition of the DIP network and strictly trained by normal clean images and used. Their neural network-based model does not require several kinds of adversarial noisy images for training. Their experimental results show that their method has a better performance compared to other detection methods in all datasets which even works well in real images as well.

### 5.2 Adversarial training

Adversarial training is one of the effective and widely used methods which successfully defend against attacks from adversarial examples up to some extent. Szegedy et al. [2] introduced first the idea of adversarial training where they trained the neural networks with the mixture of adversarial examples and clean examples so that model can learn to classify correctly between them. Thereafter, Goodfellow et al. [3] introduced FGSM to produce adversarial training to enhance the robustness of the model.

Kurakin et al. [4] further extended Goodfellow et al. [3] work and trained a model using a single-step attack that was robust to single-step perturbations but does not work well in multi-step attacks.

Thereafter, Qin et al. [15] used local linearity regularization (LLR) instead of FGSM based regularization method and find that models which have smaller local linearity values are more robust.

Huang et al. [37] introduced a new method based on the min-max formulation where the machine learning model learns from a strong adversary. Their experiment shows that their proposed method minimizes the classi-

fication error and maximizes the robustness of the target model. Madry et al. [5] introduced a projected gradient descent (PGD) based attack that significantly increased the robustness of neural networks and provide the security guarantee protection from any adversary. However, when trained to large datasets, the computational complexity becomes too high.

Shafahi et al. [38] understand this problem and proposed an algorithm that reduces the overhead of generating adversarial examples through recycling gradient information computed during the process of updating model parameters. They also found that their algorithm was 7 to 30 times faster and achieves better robustness when compared to PGD adversarial training on large datasets.

Wong et al. [39] goes further and proposed to combine both FGSM and random initialization which is as effective as PGD of Shafahi et al. [38] method with faster training of model but Andriushchenko and Flammarion [40] investigated that both Shafahi et al. [38] and Wong et al. [39] methods have some overfitting. They also found that when the random initialization is used then it also decreases the magnitude of perturbations value. Therefore, introduced a new regularization method called GradAlign that prevents catastrophic overfitting and enhance the FGSM as well.

### 5.3 Certified Training

It is worth considering the case when the model is trained using certified training data rather than normal adversarial training. Although adversarial training has been recognized as a powerful defense method against adversarial attacks. However, it has been observed by Li et al. [21] that adversarial training also cause overfitting. Therefore, they introduced a framework that enables a certifiable lower bound to be applied to the prediction accuracy against adversarial examples. Their experiments describe the effectiveness of the attack method along with the significant improvements in the defense.

Using a semidefinite relaxation, Raghunathan et al. [22] propose a method that produces a certificate that is valid for a given network and test input where error can not be forced to exceed a certain value by an attacker. They also optimize the certificate together with the network parameters to provide an adaptive regularizer in order to make the model robust against all kinds of attacks.

For the first time, Lecuyer et al. [23] present a certified defense for any arbitrary model type that scales well both to large networks and datasets. Their Pixel Differential Privacy (PixelDP) defense provides a rigorous, generic, and flexible foundation for defense.

### 5.4 GAN Defense

The GAN is a well-known model for defending against adversarial attacks. Goodfellow et al. [24] introduced a Generative Adversarial Network (GAN) framework. The generative model captures the data distribution and the discriminative model analyze the probability of whether the data came from the training dataset or from the generative model. Here, if the generative model gets better then the discriminative model will get worse and vice versa. The main idea of this framework is that the min-max game is played between the discriminator and generator where the generator attempts to misguide the discriminator and discriminator main purpose is to not get fooled by the generator. Therefore, both generator and discriminator models get trained well. Finally, the generator is considered for generating data for the model. As shown in Figure 3, there are two neural networks, the generator and discriminator. The generator takes the random noise vector as an input and then uses the convolutional network layer to create a fake image that is almost identical to the original, and then it goes to the discriminator, where it determines whether it is real or fake, and then it applies the loss through discriminator so that it can be better classified later, similarly, the generator also learns through the same loss function, and it continues until both become excellent at their job.

Xiao et al. [25] first introduced an AdvGAN method with generative adversarial networks (GANs) for the generation of adversarial examples that learns and estimate the distribution of unperturbed instances. They used AdvGAN in both semi-white box and black-box attack and their experimental result show that their defenses mechanism have a comparatively high success rate when adversarial examples were generated by AdvGAN

Samangouei et al. [26] introduced a method Defense-GAN that can be used with any kind of classification model for defense against adversarial attack. They used the Wasserstein Generative Adversarial Network (WGAN) [28]. model for the generative model due to the stability of the training method. For learning the distribution of unperturbed training data, WGAN uses Wasserstein loss, unlike min-max loss which is used by GAN. Defense-GAN provides protection from both white-box and black-box adversarial attacks.

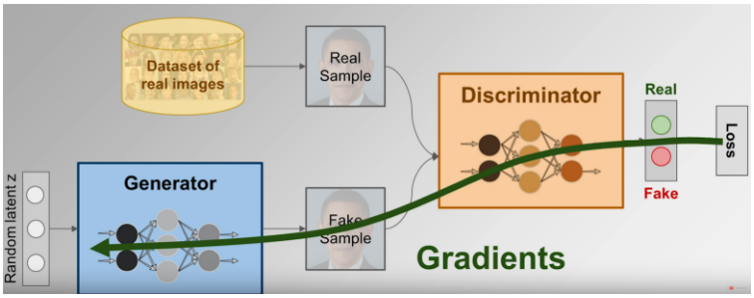

Figure 3: Face editing with Generative Adversarial Networks by Arxiv Insights. [41]

Song et al. [27] proposed a method where they used conditional generative models for generating adversarial examples where they train the model through an Auxiliary Classifier Generative Adversarial Network (AC-GAN) [29] for the data distribution for each class. Thereafter, they try to identify misclassified data generated under the generative model in AC-GAN conditioned on the desired class. They found that generated unrestricted adversarial examples belong to the desired class. Their experimental results indicate that unrestricted adversarial examples are highly effective in adversarial attacks compared to the traditional adversarial training and certified defense methods.

### 5.5 Comparative Analysis of Defense Method

Below you will find the most popular defense methods used to protect the model and make them robust against adversaries along with some of the interesting facts which have been observed by the researcher during their experiments. In addition, this table shows how the proposed method is superior to the previously proposed method.

Table 2: Defense Method

| Method Name | Author | Remarks |
|---|---|---|
| Statistical tests | Grosse et al. [16] | Detects the adversarial examples, not effective for secondary adversarial attack |
| Steganalysis | Liu et al. [11] | Accurately detect adversarial examples, can not apply directly to secondary adversarial attack |
| LID | Ma et al. [17] | LID value of adversarial examples is much higher than the LID of original data |
| FS | Xu et al. [18] | Reduce the search space available to an adversary |
| SID | Tian et al [19] | Better performance and generalization capabilities than LID and FS |
| SFAD | Aldahdooh et al [20] | Better performance than the state-of-the-art algorithms |
| DIP | Sutanto et al [36] | Better performance compared to all in all datasets |
| LLR | Qin et al. [15] | Smaller local linearity values are more robust |
| PGD | Madry et al. [5] | Significantly increased the robustness but computation time is high for large datasets |
| GradAlign | Andriushchenko et al [40] | Does not have overfitting than [38] but have a similar result e.g. 7 to 30 times faster and achieves better robustness than PGD. |
| PixelDP | Lecuyer et al. [23] | Scales well both to large networks and datasets. |
| AdvGAN | Xiao et al. [25] | High success rate in defense |
| Defense GAN | Samangouei et al. [26] | Used with any kind of classification model, protect from both white-box and black-box adversarial attacks |
| AC-GAN | Song et al. [27] | Highly effective in adversarial attacks compared to the traditional adversarial training and certified defense methods |

# 6 DISCUSSION AND CONCLUSION

## 6.1 Discussion

It is true that adversarial examples have made significant achievements in generation and defense algorithms for unstructured data, but there are still many key issues that remain. As we have seen different white-box [8] attacks such as gradient-based optimization [2, 3, 4] and higher versions of them to improve performance and robustness, where PGD [5] methods have been considered superior to previously proposed methods. We have also seen that gradient-free methods (ATNs) [6] have better computation. As part of the study, we looked at different black-box methods [12, 13, 32, 33, 34, 14] where attackers observed the output and created parallel models by querying several times to the machine learning model. We found that the DaST [14] method has higher performance compared to other methods including FGSM, BIM, PGD and C&W in both targeted and non-targeted attacks. But DaST cannot generate adversarial examples on its own and must be used with gradient-based attack methods. Hence, it remains an open problem to create new methods similar to DaST that can generate direct attacks. In addition, there are other methods such as the SWITCH method which is a highly query-efficient black-box adversarial attack model that achieves state-of-the-art performance. Simulation Attack [42] reduces the complexity of computing of any target model and accurately mimics any unknown model. There is still much that needs to be researched and a more suitable method of attack could be found.

Furthermore, it has also been shown how different defense methods can be used to protect a model, whether the attack is detected in the initial stage [16, 11, 17, 18, 19, 20, 36] or when creating the model with a mixture of adversarial examples and clean examples [2, 3, 4, 15, 37, 5, 38, 39, 40] or when training it with certified datasets. In addition, we studies several GAN methods [24, 25, 26, 28, 27] where we found that AC-GAN [29] is highly effective in adversarial attacks compared to state-of-the-art methods. Although, none of the proposed algorithms are promising which we can completely rely on to defend against adversarial examples. GAN training is a research topic that is ongoing, however, the combination of two methods and application of different techniques may improve its robustness and ability to protect from adversaries.

## 6.2 Conclusion

We have seen several different adversarial examples generation attack methodology and defense algorithms. We studied how and why did the attackers succeed to achieve their goal. Although defense algorithms try to protect the system from adversaries to some large extent but still have many key issues that remain an open problem to be solved. We studied the best available defense methods and in which scenario we could use them. Therefore, ensuring the robustness and performance of the machine learning model and at the same time avoiding an attack from the adversarial example is an important area of research that has much space to be worked on further. We hope that in future using some new methodology or combining two different methods may improve the machine learning model and maximize accuracy by protecting the adversarial examples.

## References


[1] Adversarial machine learning, 2021. URL https://en.wikipedia.org/wiki/Adversarial_machine_learning.

[2] Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow I.J., and Fergu R. Intriguing properties of neural networks, 2014. URL http://arxiv.org/abs/1312.6199.

[3] Goodfellow I.J., Shlens J, and Szegedy C. Explaining and harnessing adversarial examples, 2014. URL http://arxiv.org/abs/1412.6572.

[4] Kurakin A., Goodfellow I.J., and Bengio S. Adversarial machine learning at scale, 2017. URL https://arxiv.org/abs/1611.01236.

[5] Madry A, Makelov A, Schmidt L, Tsipras D, Adrian V. Towards deep learning models resistant to adversarial attacks, 2017. URL https://arxiv.org/abs/1706.06083.

[6] Baluja S, Fischer I. Learning to attack: Adversarial transformation networks, 2018. URL https://research.google/pubs/pub46527/.

[7] Papernot, N., Mcdanierl, P., Jha, S., Fredrikson, M., Celik, Z. B., AND Swami, A. The limitations of deep learning in adversarial settings, 2015. URL https://arxiv.org/abs/1511.07528.

[8] Carlini N, Wagner D. . Towards evaluating the robustness of neural networks, 2017. URL https://arxiv.org/abs/1608.04644.

[9] Dalvi N, Domingos P, Sanghai S, Verma D. *Adversarial classification. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 2004.

[10] Biggio A, Corona I, Maiorca D, Nelson B, Srndic N, Laskov P, Giacinto G, Roli F. Evasion attacks against machine learning at test time, 2017. URL https://arxiv.org/abs/1708.06131.

[11] Liu J, Zhang W, Zhang Y, Hou D, Liu Y, Zha H, et al. Detection based defense against adversarial examples from the steganalysis point of view, 2019. URL https://arxiv.org/abs/1806.09186.

[12] Chen P-Y, Zhang H, Sharma Y, Yi J, Hsieh C-J. Zoo. Zeroth order optimization based black-box attacks to deep neural networks without training substitute models., 2017. URL https://arxiv.org/abs/1708.03999.

[13] Tu C-C, Ting P, Chen P-Y, Liu S, Zhang H, Yi J, et al. Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks., 2019. URL https://arxiv.org/pdf/1711.00123.pdf.

[14] Zhou M, Wu J, Liu Y, Liu S, Zhu C. Dast: Data-free substitute training for adversarial attacks, 2020. URL https://arxiv.org/abs/2003.12703.

[15] Qin C, Martens J, Gowal M, Krishnan D, Dvijotham K,Fawzi A, De S, Stanforth R, and Kohli P. Adversarial robustness through local linearization, 2019. URL https://arxiv.org/abs/1907.02610.

[16] Grosse, K., Manoharan, P., Papernot, N., Backes, M.,McDaniel, P. On the (statistical) detection of adversarial examples, 2017. URL https://arxiv.org/abs/1702.06280.

[17] Ma, X., Li, B., Wang, Y., Erfani, S.M., Wijewickrema, S., Schoenebeck, G., Bailey, J., et al. Characterizing adversarial subspaces using local intrinsic dimensionality, 2018. URL https://arxiv.org/abs/1801.02613.

[18] Xu, W., Evans, D., Qi, Y., Wijewickrema, S., Schoenebeck, G., Bailey, J., et al. Feature squeezing: Detecting adversarial examples in deep neural networks, 2017. URL https://arxiv.org/abs/1704.01155.

[19] Tian J, Zhou J, Li Y, Jia D. Detecting adversarial examples from sensitivity inconsistency of spatial-transform domain., 2021. URL https://arxiv.org/abs/2103.04302.

[20] Aldahdooh, A., Hamidouche, W., D´eforges, O. Revisiting model's uncertainty and confidences for adversarial example detection, 2021. URL https://arxiv.org/abs/2103.05354.

[21] Li B, Chen C, Wang W, and Carin L. Second-order170adversarial attack and certifiable robustness, 2018. URL https://www.arxiv-vanity.com/papers/1809.03113/.

[22] Raghunathan, A., Steinhardt, J., and Liang P. Certified defenses against adversarial examples. international conference on learning representations, 2018. URL https://openreview.net/forum?id=Bys4ob-Rb.

[23] Lecuyer M, Atlidakis V, Geambasu R, Hsu D, and Jana S. Certified robustness to adversarial examples with differential privacy, 2018. URL https://arxiv.org/abs/1802.03471.

[24] Goodfellow I.J., Abadie J.P., Mirza M., Xu B, Farley D.W., Ozair S, Courville A, Bengio Y, . Generative adversarial networks, 2014. URL https://arxiv.org/abs/1406.2661.

[25] Xiao C, Li B, Zhu JY, He W, Liu M, Song D. Generating adversarial examples with adversarial networks, 2018. URL https://arxiv.org/abs/1801.02610.

[26] Samangouei P, Kabkab M, and Chellappa R. Defense-gan: Protecting classifiers against adversarial attacks using generative model, 2018. URL https://arxiv.org/abs/1805.06605.

[27] Song Y, Shu R, Kushman N, Ermon S. Constructing unrestricted adversarial examples with generative models., 2018. URL https://arxiv.org/abs/1805.07894.

[28] Arjovsky M., Chintala S., Bottou L. Wasserstein gan, 2017. URL https://arxiv.org/abs/1701.07875.

[29] Odena A, Olah C, Shlens J. Conditional image synthesis with auxiliary classifier gans, 2016. URL https://arxiv.org/abs/1610.09585.

[30] Eykholt K, Evtimov I, Fernandes E, Li B, Rahmati A, Xiao C, Prakash A, Kohno T, Song T. Robust physical-world attacks on deep learning models, 2017. URL https://arxiv.org/abs/1707.08945.

[31] Yang J, Jiang Y, Huang X, Ni B, Zhao N. Learning black-box attackers with transferable priors and query feedback, 2020. URL https://arxiv.org/pdf/2010.11742.pdf.

[32] Chen J, Michael I.J., Martin J.W. A query-efficient decision-based attack., 2020. URL https://arxiv.org/pdf/1904.02144.pdf.

[33] Papernot N, McDaniel P, Goodfellow I, Jha S, Celik ZB, Swami A. Practical black-box attacks against machine learning, 2017. URL https://arxiv.org/abs/1602.02697.

[34] Seong Joon Oh, Max Augustin, Bernt Schiele, and Mario Fritz. Towards reverse-engineering black-box neural networks, 2018. URL https://arxiv.org/pdf/1711.01768.pdf.

[35] N. Papernot, P. McDaniel, and I. Goodfellow. Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples, May 2016. URL https://arxiv.org/abs/1605.07277.

[36] Evan Sutanto R and Lee S. Real-time adversarial attack detection with deep image prior initialized as a high-level representation based blurring network, 2021. URL https://www.mdpi.com/2079-9292/10/1/52/pdf.

[37] Huang R, Xu B, Schuurmans D, and Szepesv´ari C. Learning with a strong adversary, 2015. URL https://arxiv.org/abs/1511.03034.

[38] Ali S, Najibi M, Amin G, Xu Z, John D, Studer C, et al. Certified robustness to adversarial examples with differential privacy, 2019. URL https://arxiv.org/abs/1904.12843.

[39] Wong E, Rice L, Kolter JZ. Fast is better than free: Revisiting adversarial training, 2020. URL https://arxiv.org/abs/2001.03994.

[40] Andriushchenko M. and Flammarion N. Understanding and improving fast adversarial training, 2020. URL https://arxiv.org/abs/2007.02617.

[41] Editing faces using artificial intelligence, 2019. URL https://youtu.be/dCKbRCUyop8.

[42] Ma C, Chen L, Jun-Hai Y. Simulating unknown target models for query-efficient black-box attacks, 2020. URL https://arxiv.org/abs/2009.00960.