Adversarial Robustness of Contemporary Machine Learning Models

Final Report

Afia Afrin
aafrin@ualberta.ca

Abstract

Recent advancements in the field of adversarial learning have demonstrated that it is possible to fool intelligent systems in a variety of ways including poisoning the training data, perturbing the test samples, and exploiting over-fitting on training data. Given that we live in an era of intelligent devices and artificial brains, it is critical to understand how adversaries affect these models. A thorough understanding of this issue would aid us in identifying model vulnerabilities and developing effective mitigation strategies. In this work we aim to leverage an open-source adversarial machine learning (AML) tool to generate various adversarial attacks against different machine learning models in different domains and analyze the model robustness.

Keywords: Adversarial attacks, Adversarial Robustness Toolbox, Robust Machine learning

1 Introduction

From *chess* to *go*, *smart homes* to *smart healthcare systems*-artificial intelligence (AI) and machine learning (ML) have been pervasive throughout the world. The vast application arena for intelligent systems begs the obvious question: how safe are these systems? A recent study by Ram Shankar Siva Kumar et al. [13] has published some interesting insights on the industrial practice of securing the ML models against adversarial attacks. Based on the interviews with 28 different organizations they conclude that, there is no precise guideline or appropriate tool to protect, detect or respond to adversarial attacks. They identified numerous possible explanations for this particular issue. One argument given is that industry practitioners, including security experts, believe that adversarial threats are too futuristic to be concerned about. Additionally, there is a widespread perception that major machine learning libraries are intelligent enough to defend themselves. The goal of this work is to examine the extent to which we can agree on these popular beliefs.

We aim to address the following research questions (RQ):

- RQ1: Do adversaries exhibit a consistent level of effectiveness across different domains?
- RQ2: How resilient are contemporary machine learning models to adversarial attacks?
- RQ3: Are adversarial attacks too far in the future for us to consider right now?

This paper is organized as follows. Section 2 introduces the necessary terminologies along with brief description of adversarial attacks and conventional defense mechanisms. Section 3 presents an overview of our workflow. Section 4 elaborates the implementation details. Section 5 presents the experimental results. Section 6 discusses the challenges we faced while using the *ART* tool. Section 7 presents a summary of prior research works in the field of adversarial machine learning and finally section 8 concludes the work.

2 Background

In this section we introduce different types of adversarial attacks, defense mechanisms, and tools that can be used to generate adversarial attacks.

2.1 Adversarial Attacks

Adversaries are threats to machine learning (ML) models. While there are various types of adversarial attacks, they all share a common goal: to fool the ML models with deceptive data. Based on the amount of information that the adversary has about the ML model, adversarial attacks can be categorized into two types- *white-box attacks* and *black-box attacks*. As the names suggest, during a *white-box attack* the adversary has internal knowledge about the ML model including the model architecture, parameter values, and weights associated with the connections. However, in a *black-box attack* the adversary has no prior knowledge about the model.

Based on the attack methodology, adversarial attacks can be categorized into two different classes- *evasion attacks* and *poisoning attacks*. *Evasion attack* raises security breach by exploiting vulnerabilities during the classification phase. Here, the test samples get manipulated by the attacker which results in model's accuracy drop. *Poisoning attack* works in a different way. Instead of targeting the classification phase, it manipulates the training samples and mess up with the learning process.

In this project, we have limited ourselves to testing the robustness of contemporary ML models against *white-box evasion* attacks. We implemented three different adversarial attacks, each of which is briefly described below.

Final Project Report, CMPUT 664, Winter 2022

Fast Gradient Sign Method (FGSM). *FGSM* is a gradient-based, white-box evasion attack, introduced by Goodfellow et. al [10]. It was primarily generated against the models that use image data. Essentially, *FGSM* computes the gradients of the loss function with respect to the input image and then uses the sign of the gradients to create a new image (i.e., the adversarial image) that maximizes the loss. We can express the whole idea with the following equation-

$$\tilde{x} = x + \epsilon \times sign(\nabla_x J(\theta, x, y) \quad (1)$$

Here,
$\tilde{x}$ = The adversarial image generated using *FGSM*
x = The original input image
y = The ground truth label for the input image x
ϵ= Hyperparameter to control the perturbation amount
θ = The ML model
J = The loss function

FGSM, also known as *the simplest* adversarial attack algorithm, is a one-step algorithm. There are several upgraded versions of *FGSM*. *PGD* is one of them, which we discuss next.

Projected Gradient Descent (PGD) . The main idea of the *PGD* [18] attack is to iteratively apply *FGSM* in order to create more sophisticated attacks. Each iteration obtains a gradient score based on ϵ-norm ball on the original image and then projects back to the original ϵ.

DeepFool . DeepFool [20] is an untargeted adversarial methodology that uses the $l2 - norm$ to produce adversarial samples with minimal perturbation. The minimal perturbation is defined as the perpendicular distance between the data point (for example: an image) and the decision boundary. There exists a closed-form formula to launch *DeepFool* against linear, binary classifiers. However, non-linear or multiclass classifiers necessitate multiple iterations to generate the attack.

2.2 Defense Mechanisms

Conventional approaches to adversarial defenses can be categorized into two different classes: *model hardening*, and *input cleansing*. In this work we implemented the *model hardening with adversarial training* method on *text* and *audio* data. In this method, the classifier is trained using an augmented training dataset which includes adversarial samples. This helps the model to recognize and correctly classify the adversarial samples during the inference phase, thereby enhancing the overall model robustness.

Input cleansing, on the other hand, employs a different technique to prevent adversarial attacks. In this method, *adversarial detectors* [3], [4] are used to ensure the sanity of the inputs to the neural network during the training and testing phase. Thus, all malicious data are discarded during the scanning phase, ensuring that adversaries can never reach the model.

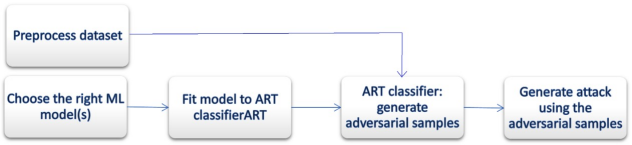


Figure 1. Extending *ART*: The Workflow Diagram

2.3 Adversarial Machine Learning Tools

Adversarial robustness toolbox (ART)[22], *AdvBox*[11], *CleverHans*[23] and *FoolBox*[24] are the widely used tools for creating adversarial attacks and implementing countermeasures against them. Another similar tool is *Deepsec*[16] but its implementation accuracy is questionable [2]. Each of these tools has their own pros and cons. For example, while *ART* is capable to generate attack against the highest number of ML libraries, *FoolBox* offers the highest range of attacks [2]. While looking for the right tool, we discovered that with plenty of resources and regular maintenance, *ART* is the best-suitable option for our project. The following section provides an overview of the tool and discusses how we used it in our work.

3 Overview

ART supports a total of 11 different ML libraries including *Python's* three most popular frameworks: *Tensorflow*, *Keras*, and *PyTorch*. This is the highest level of compatibility that contemporary AML tools offer across different libraries [2]. In addition, it supports 19 adversarial attacks. However, extending the attacks to diverse domains requires additional efforts including rigorous preprocessing of the dataset, fitting the ML models to the *ART* classifiers, and collecting the final results in proper formats. This indicates that when extending the adversarial attacks in a new domain using *ART*, we need to follow some basic steps to ensure compatibility and accurate implementation. Fig. 1 depicts the generic workflow to extend the *ART* adversarial attacks. Detailed description of the workflow is presented in section 4.

4 Implementation Details

This section elaborates the implementation details by explaining how we have addressed each of the research questions outlined in section 1.

RQ1: Adversaries are widely recognized as a threat to image classification models and computer vision applications. However, their effectiveness in other domains, particularly with time series data (such as audio or network traffic data), has not been thoroughly studied yet.

In this project, we extended the features offered by *Adversarial robustness toolbox* to deploy adversarial attacks against image, text, and audio dataset. We have utilized the following datasets:

Table 1. Adversarial Attacks on Image Dataset

Dataset	Model	Attacks	Model Accuracy on Benign Data (%)	Model Accuracy on Adversarial Data (%)
Fruits-360	CNN	FGSM	91.70	22.54
		DeepFool	91.70	20.55
	Sequential NN	FGSM	97.17	28.82
		PGD	97.17	16.00
MNIST	Sequential NN	FGSM	94.75	2.19
		DeepFool	94.75	1.95

- Image data: **Fruits 360** [21], **MNIST** [14]
- Text data: **Smart Grid Stability** dataset [1]
- Audio data: **UrbanSound8K** dataset [25]

As mentioned in the previous section, datasets need to be preprocessed before being fed into an *ART* classifier. There is no general rule to preprocess data, it depends on what type of data and which *ART* classifier we are using. For example, when working with image data, the pixel values should be in a scale from 0-255 for the attacks to be successful. Another such example is, for the dataset to be compatible with *ART's KerasClassifier*, we need to increase its dimension by 1. However, *PytorchClassifer* does not need any such modification.

While working with text data, the class labels need to be encoded either manually or by using *labelencoder*. Then, if it is a multi-class classification problem, then the labels need to be converted into one-hot-encoded vectors.

We leveraged *Python's* audio library *Librosa* for preprocessing the audio dataset. To extract the features from audio files, we used the *MFCC* algorithm. Introduced by Davis and Mermelstein [6], this algorithm has been widely used in automatic speech and speaker recognition since 1980. This algorithm allows us to convert the audio files into a series of feature arrays, containing components of the audio signal that are good for identifying the linguistic content.

RQ2: To analyze how robust the libraries are, we have deployed a series of adversarial attacks in two different Python libraries, namely: *Tensorflow.Keras*, and *PyTorch*. Unfortunately, both of them have been found vulnerable to adversarial attacks which indicates that the popular belief about the inherent robustness of widely used ML libraries is not true.

RQ3: Our experimental findings show that, with appropriate modification, any adversarial attack algorithm can be adopted for diverse domains and multiple ML libraries. Thus, without any doubt, we can conclude that the notion that *"adversarial attacks are futuristic"* is a widely spread misconception.

5 Experimental Results

We have implemented three different adversarial attacks against two popular ML libraries on four different datasets. This section discusses the experimental results by addressing the first two research questions.

5.1 Adversarial attacks in different domains (RQ1)

In order to address this question we present the results that have been obtained by launching adversarial attacks on diverse domains. The following three subsections discuss the effects of adversarial attacks on image, text, and audio data, respectively.

5.1.1 Adversarial attacks on image data. We have analyzed the effect of the *FGSM*, *DeepFool*, and *PGD* attacks against two distinct image classifiers. As per the experimental results, all of the attacks have a detrimental effect on model performance. Additionally, in order to determine whether attack strength is dataset-dependent or not, we conducted the experiment on two distinct datasets. Interestingly, we discovered that datasets do indeed have an effect on attack strength. For example, when working with the *Fruits-360* dataset, *FGSM* reduces model accuracy by 70% on average. However, the same attack is more effective against the *MNIST* dataset, where accuracy is reduced by approximately 97%. A simple observation at both of the datasets explains this behavior. While working with *MNIST* dataset, it is easier to fool the model by altering a few pixels in the input image. It does not require much effort to pollute a handwritten `0` in such a way that the model thinks it is a 5, or manipulating 4 in such a way that it gets classified as 9. However, this is not true for the *fruits-360* dataset. We need substantial perturbation to modify the image of a red apple so that it gets labeled as a banana, which is different in both- color and shape. Table 1 summarizes the experimental findings of this experiment.

While *DeepFool* aims to cause as little perturbation to the test data as possible so that the ML model gets fooled but no difference is visible to human eyes, for the other two attacks (*FGSM* and *PGD*), we can control the amount of perturbation. Increasing the noise level reduces the model's accuracy further, but results in blurry and unreadable images. Fig. 2 shows the original image and different adversarial images of an apple that have been generated by applying the *FGSM* attack with different perturbation values. Fig. 3 illustrates how increasing the perturbation amount of the *FGSM* attack

Table 2. Adversarial attacks on text dataset

Attack	Model accuracy on benign data (%)	Model accuracy on adversarial data (%)	Accuracy drop(%)
FGSM (ϵ = 0.2)	99.2	64.37	35
PGD (ϵ = 0.2)	99.2	64.37	35

exponentially decreases model accuracy. Similar results can be obtained by generating *PGD* attack with different values of ϵ.

Fig. 4 illustrates the comparison between the original image and three adversarial images generated by *FGSM*, *DeepFool*, and *PGD* respectively. As can be seen, *FGSM* and *PGD* work similarly. This is nothing surprising, since *PGD* is just an upgraded version of *FGSM*. However, the *DeepFool* attack is the best among these three. It produces the minimum possible perturbation while generating the strongest possible attack. This is the reason we see no visible differences between the original image and the adversarial image generated by *DeepFool* as shown in fig. 4.

5.1.2 Adversarial attacks on text data.

ART offers limited attacks against binary classifiers. Hence, we were able to generate only two distinct adversarial attacks, *FGSM* and *PGD*, on the *smart grid stability*[1] data. For this experiment, we used the simple sequential artificial neural network from *tensorflow*. Both of these attacks are capable of reducing model accuracy significantly in a very short period of time. Just like the previous experiment, both attacks behave similarly against a particular model. Since *PGD* is just an upgraded version of *FGSM*, this is not surprising.

To quantify the noise introduced by the attacks into the data sample, we wrote a simple Python script that shows the average perturbation added to each of the features. We define perturbation by the *l2-norm*. For each of the 13 features in the Smart Grid Stability dataset [1] dataset, average perturbation is defined as:

$$\frac{1}{m}\sqrt{(F_1^i - \tilde{F}_1^i)^2 + (F_2^i - \tilde{F}_2^i)^2 + (F_3^i - \tilde{F}_3^i)^2 + ... + (F_m^i - \tilde{F}_m^i)^2}$$

Here,
m = total number of test samples. For this dataset, m = 6000.

F_k^j = j^{th} feature of the k^{th} sample. For this particular dataset, 1 <= j <= 13 and 1 <= k <= 6000.

$\tilde{F}_k^j$ = j^{th} feature of the k^{th} adversarial sample. Again, 1 <= j <= 13 and 1 <= k <= 6000.

Table 3. Effect of PGD hardening defense on model accuracy

Dataset	Defense	Attack	Robust model accuracy (%)	
			on benign data	on adversarial data
Text	PGD hardening	FGSM	95.62	74.70
		PGD	95.62	89.88
Audio		FGSM	92.98	81.88
		PGD	92.98	92.60

Fig. 5 depicts the average perturbation added to the features by two adversarial attacks, FGSM (ϵ = 1) and PGD (ϵ = 1).

Finally, to gain a better understanding of the amount of perturbation, we identified the data sample that has been altered the most. Fig. 6 shows the maximum amount of noise that have been added by *FGSM* and *PGD* respectively. As the figures show, the amount of noise added is too small to be noticabe to human eyes. However, as shown in table 2, this slightest data alteration results into significant accuracy drop in ML models.

5.1.3 Adversarial attacks on audio data.

The *DeepFool* attack proposed by Moosavi-Dezfooli et al. [20] is capable of deceiving machine learning models significantly with the least amount of data modification. As fig. 4 depicts, our experimental findings also supports this claim. Therefore, we extended the experiment to the audio domain by implementing this attack on the UrbanSound8K dataset [25].

As mentioned in Section 4, we extracted features from audio data using the *MFCC* algorithm. The features were then saved as a comma-separated values file that was later used as input for a sequential ANN. After 100 epochs of training, the ANN achieved a validation accuracy of 82.62%. Then we used the *DeepFool* attack to generate adversarial data samples. Testing on these adversarial samples results in a dramatic decrease in model accuracy of approximately 78%, resulting in a model accuracy of only 17.93%. However, the added noise is so insignificant that the adversarial sample sounds almost identical to the original. The waveforms of an original audio file and of the corresponding adversarial sample are shown in fig. 7.

5.2 Adversarial attacks against popular ML libraries (RQ2)

We generated the above mentioned attacks against *Python's PyTorch* library and *tensorflow.keras* API which uses *tensorflow* as its backend. As demonstrated in the preceding section, all of the attacks successfully penetrated both libraries which indicates that even the most widely used machine learning

(a) Original image (label: apple, accuracy= 91.33%) (b) FGSM ($\epsilon = 0.005$) (labeled as: apple, accuracy= 88.06%) (c) FGSM ($\epsilon = 0.01$) (labeled as: apple, accuracy= 87.53%) (d) FGSM ($\epsilon = 0.1$) (labeled as: cherry, accuracy= 26.88%) (e) FGSM ($\epsilon = 0.2$) (labeled as: avocado, accuracy= 22.54%)

Figure 2. Adversarial samples generated by applying FGSM attack with different perturbations. Image **a** is an original image from the 'apple' class, images **b-e** are the generated adversarial samples.

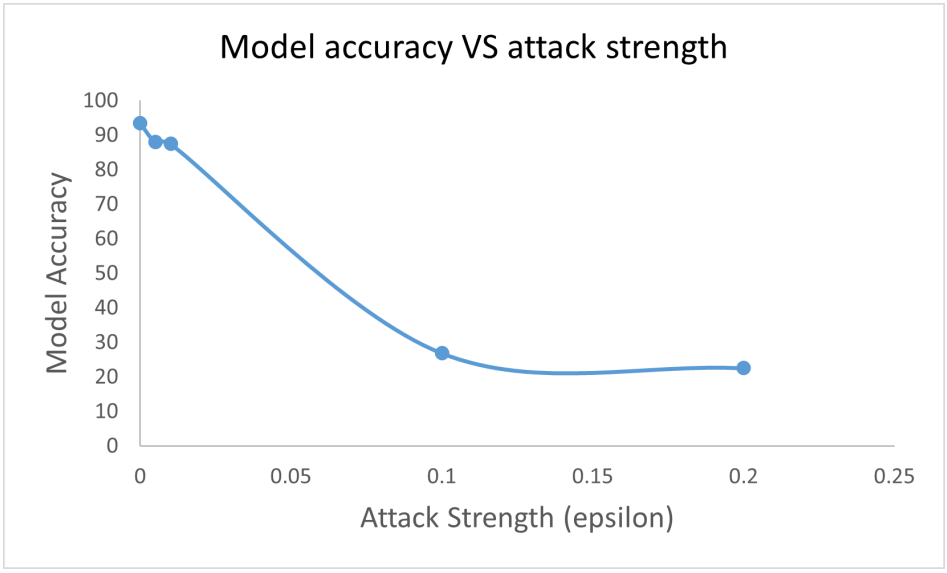


Figure 3. Model accuracy VS attack (FGSM) strength curve.

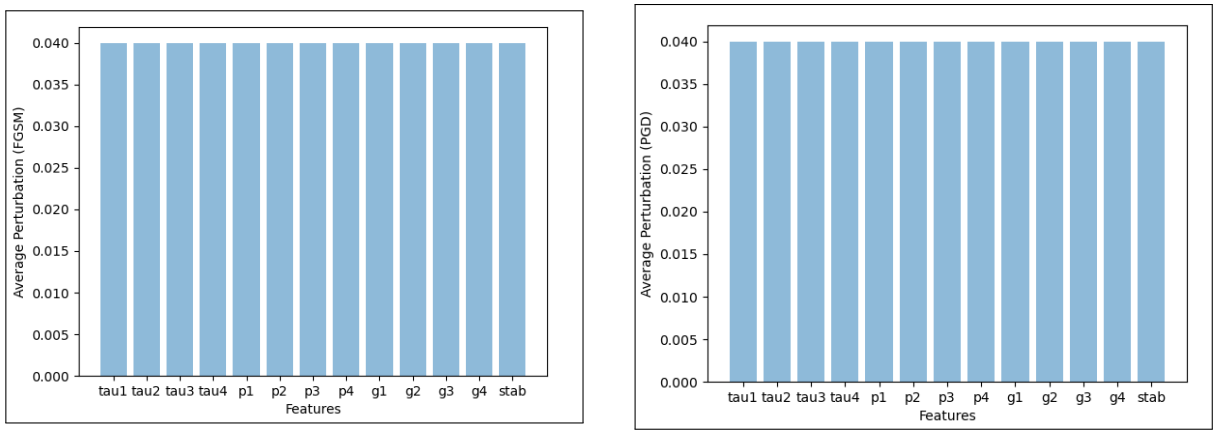
(a) Original Image (label: apple, accuracy: 91.70%) (b) FGSM ($\epsilon = 0.1$) (labeled as: cherry, accuracy: 26.88%) (c) DeepFool (labeled as: avocado, accuracy: 20.55%) (d) PGD ($\epsilon = 5$) (labeled as: banana, accuracy= 16%)

Figure 4. Adversarial samples generated by applying different attacks. Image **a** is an original image from the 'apple' class, images **b-d** are the generated adversarial samples.

libraries are vulnerable to adversarial attacks. This necessitates additional research into the development of robust and secure machine learning models.

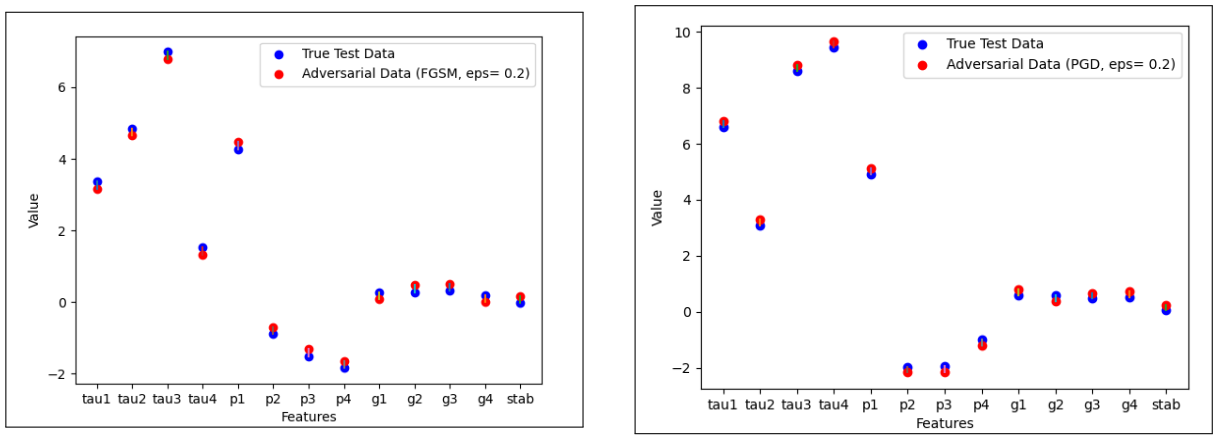
5.3 Defense Mechanism

To defend against adversarial attacks, we implemented a widely used defence technique called *model hardening with adversarial training*. Adversarial training [10] enhances the



(a) FGSM ($\epsilon = 0.2$) (b) PGD ($\epsilon = 0.2$)

Figure 5. Average perturbation added to the text dataset by different adversaries.



(a) FGSM ($\epsilon = 0.2$) (b) PGD ($\epsilon = 0.2$)

Figure 6. Attacks on text data: maximum amount of noise that has been added by different adversaries.

classifier's robustness metric by augmenting adversarial samples with the training data. This is a relatively simpler and faster method that has been demonstrated to be effective against a variety of adversarial attacks in previous works [27], [19]. However, there are some limitations: (i) while training a model with adversarial samples improves its robustness against specific adversarial attacks, it does not guarantee an overall increase in robustness, (ii) Model hardening is not an effective strategy for defending the model against *black-box* attacks. [2].

We tested the effectiveness of model hardening on each of the implemented models on text and audio data domain. To conduct this experiment we trained the models with a mixture of normal training samples and *PGD* hardened training samples. As expected, adversarial-trained models helps to enhance model accuracy on adversarial data. Table 3 summarizes the results from this experiment.

6 Discussion

Adversarial machine learning tools, such as: *ART* or *AdvBox*, have made it easier to work with adversarial attacks and corresponding mitigation approaches. However, extending the functionalities of these tools and implementing attacks on diverse domains require a good amount of additional works. The adversarial classifiers in *ART* have been defined in a very restrictive manner and they do not support all kind of ML models. Moreover, there are some limitations of this tool. For example, *ART* has no provision to implement the *DeepFool*, *ZooAttack*, or *Carlini and Wagner (CW)* attack against binary classifiers even though all these attack models support generating attacks against binary classifiers. However,

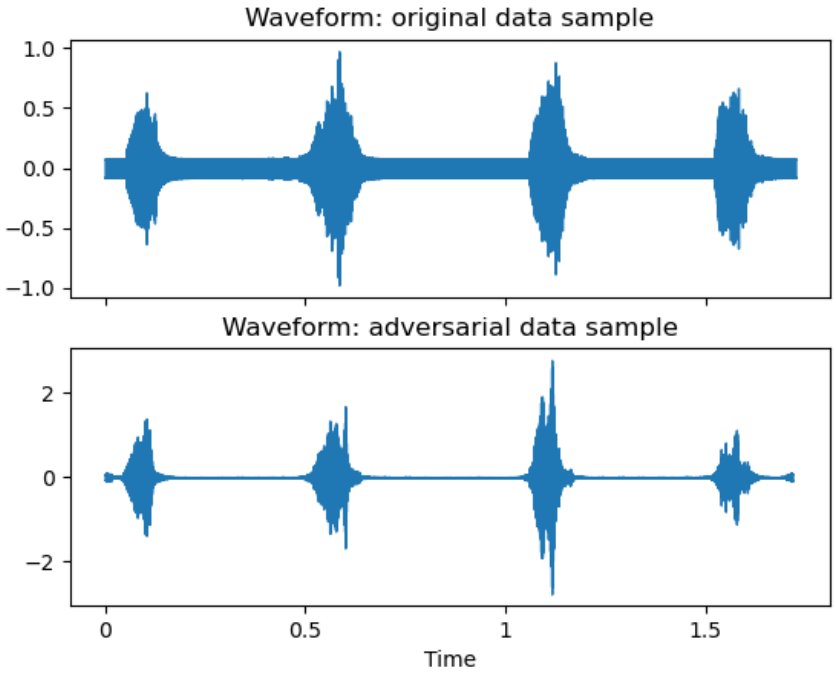


Figure 7. Waveforms of original and adversarial audio data.

these tools are continuously being upgraded to overcome these limitations and to accommodate more features.

7 Related Work

Asha et al. [2] conducted an in-depth investigation of the variety of adversarial attacks, defenses, and metrics against two different ML models. They analyzed three open-source AML tools, namely: *Foolbox*, *ART*, and *CleverHans*. While there are several other tools capable of generating diverse attacks, it is worth noting that the majority of earlier works relied on only a few. This is because not all the tools offer the same level of accuracy and reliability. To mention a few, *ART* has been used to develop novel adversarial image detection methods [17], to improve the robustness of adversarial training[9], and to detect tainted data via clustering [5]. Leveraging *ART*, *Foolbox*, and *CleverHans*, Serban et al. [26] provided a characterisation of the phenomenon of adversarial examples. To facilitate further research Yinpeng Dong et al. [7] established a comprehensive benchmark to evaluate adversarial robustness. Interestingly, all of these works are contained within the same domain. Their sphere of interest is restricted to image dataset exclusively. For example, CIFAR10 [12], the image dataset containing 60,000 colour images of ten different objects, has been utilized in several prior works [14], [2] [17] [5], [7]. Similarly, the affects of adversarial attacks on the handwritten digits database, MNIST [14], have been analyzed in some literature [2], [9].

Duan et al. [8] present one of the most comprehensive works in this domain. They evaluated twenty (20) white-box based black-box based open source adversarial attack tools and successfully implemented a robust, black-box based adversarial attack capable of breaking the HGD defense [15] with 95% success rate. However, it is restricted to discretization problems and untargeted attacks only.

To summarize, recent studies have shown that adversarial attacks pose a significant threat to computer vision and image classification models. However, the transferability of these attacks across different domains is still an open issue that needs to be researched further [2]. In this work, we aim to extend the analysis to diverse domains incorporating text and audio data.

8 Conclusion and Future Work

While adversaries are a proven threat against the applications in computer vision and image processing, their efficacy in other domains is still understudied. Conventional *AML tools* provide variety of options to generate adversarial attacks in the *image* domain. However, extending their features in other domains require rigorous experimentation. In this work, we have implemented three different adversarial attacks in three distinct domains (image, text, and audio data). Our implementation provides an easily extendable approach to extend the features of *ART*. The necessary codes to reproduce the work along with the experimental results are available on our github repository.

Conventional mitigation strategies, such as: *model hardening* or *input cleansing*, are insufficient to protect against all types of adversarial attacks. New and stronger adversaries are constantly emerging, posing threats to the machine learning applications. Analyzing model vulnerabilities and formulating more generic defense mechanisms have become a top priority for ensuring the secure implementation of intelligent systems.

References

[1] Vadim Arzamasov. 2019. Smart Grid Stability. https://www.kaggle.com/pcbreviglieri/smart-grid-stability. [Online; accessed 04-February-2022].

[2] S Asha and P Vinod. 2022. Evaluation of adversarial machine learning tools for securing AI systems. *Cluster Computing* 25, 1 (2022), 503–522.

[3] Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, and Jaehoon Amir Safavi. 2017. Mitigating poisoning attacks on machine learning models: A data provenance based approach. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*. 103–110.

[4] Marco Barreno, Blaine Nelson, Anthony D Joseph, and J Doug Tygar. 2010. The security of machine learning. *Machine Learning* 81, 2 (2010), 121–148.

[5] Bryant Chen, Wilka Carvalho, Nathalie Baracaldo, Heiko Ludwig, Benjamin Edwards, Taesung Lee, Ian Molloy, and Biplav Srivastava. 2018. Detecting backdoor attacks on deep neural networks by activation clustering. *arXiv preprint arXiv:1811.03728* (2018).

[6] Steven Davis and Paul Mermelstein. 1980. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE transactions on acoustics, speech, and signal processing* 28, 4 (1980), 357–366.

[7] Yinpeng Dong, Qi-An Fu, Xiao Yang, Tianyu Pang, Hang Su, Zihao Xiao, and Jun Zhu. 2020. Benchmarking adversarial robustness on image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 321–331.

[8] Yuchao Duan, Zhe Zhao, Lei Bu, and Fu Song. 2019. Things you may not know about adversarial example: A black-box adversarial image attack. *arXiv preprint arXiv:1905.07672* 5 (2019).

[9] Evelyn Duesterwald, Anupama Murthi, Ganesh Venkataraman, Mathieu Sinn, and Deepak Vijaykeerthy. 2019. Exploring the hyperparameter landscape of adversarial robustness. *arXiv preprint arXiv:1905.03837* (2019).

[10] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014).

[11] Dou Goodman, Hao Xin, Wang Yang, Wu Yuesheng, Xiong Junfeng, and Zhang Huan. 2020. Advbox: a toolbox to generate adversarial examples that fool neural networks. *arXiv preprint arXiv:2001.05574* (2020).

[12] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. 2014. The CIFAR-10 dataset. *online: http://www. cs. toronto. edu/kriz/cifar. html* 55, 5 (2014).

[13] Ram Shankar Siva Kumar, Magnus Nyström, John Lambert, Andrew Marshall, Mario Goertzel, Andi Comissoneru, Matt Swann, and Sharon Xia. 2020. Adversarial machine learning-industry perspectives. In *2020 IEEE Security and Privacy Workshops (SPW)*. IEEE, 69–75.

[14] Yann LeCun. 1998. The MNIST database of handwritten digits. *http://yann. lecun. com/exdb/mnist/* (1998).

[15] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, and Jun Zhu. 2018. Defense against adversarial attacks using high-level representation guided denoiser. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1778–1787.

[16] Xiang Ling, Shouling Ji, Jiaxu Zou, Jiannan Wang, Chunming Wu, Bo Li, and Ting Wang. 2019. Deepsec: A uniform platform for security analysis of deep learning model. In *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 673–690.

[17] Gabriel R Machado, Eugênio Silva, and Ronaldo R Goldschmidt. 2019. A non-deterministic method to construct ensemble-based classifiers to protect decision support systems against adversarial images: a case study. In *Proceedings of the XV Brazilian Symposium on Information Systems*. 1–8.

[18] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083* (2017).

[19] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. 2015. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644* (2015).

[20] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. 2016. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2574–2582.

[21] Horea Mureşan and Mihai Oltean. 2017. Fruit recognition from images using deep learning. *arXiv preprint arXiv:1712.00580* (2017).

[22] Maria-Irina Nicolae, Mathieu Sinn, Minh Ngoc Tran, Beat Buesser, Ambrish Rawat, Martin Wistuba, Valentina Zantedeschi, Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, et al. 2018. Adversarial Robustness Toolbox v1. 0.0. *arXiv preprint arXiv:1807.01069* (2018).

[23] Nicolas Papernot, Fartash Faghri, Nicholas Carlini, Ian Goodfellow, Reuben Feinman, Alexey Kurakin, Cihang Xie, Yash Sharma, Tom Brown, Aurko Roy, et al. 2016. Technical report on the cleverhans v2. 1.0 adversarial examples library. *arXiv preprint arXiv:1610.00768* (2016).

[24] Jonas Rauber, Wieland Brendel, and Matthias Bethge. 2017. Foolbox: A python toolbox to benchmark the robustness of machine learning models. *arXiv preprint arXiv:1707.04131* (2017).

[25] Justin Salamon, Christopher Jacoby, and Juan Pablo Bello. 2014. A dataset and taxonomy for urban sound research. In *Proceedings of the 22nd ACM international conference on Multimedia*. 1041–1044.

[26] Alexandru Constantin Serban, Erik Poll, and Joost Visser. 2018. Adversarial examples-a complete characterisation of the phenomenon. *arXiv preprint arXiv:1810.01185* (2018).

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