12/8/2023
CIS*4780
Dr. Rozita Dara

Adversarial Attacks - Exploring the Arms Race of Machine Learning

Brycen Wright, Luc Dubé

Abstract

Adversarial attacks pose a significant threat to Convolutional Neural Networks (CNNs) and their ability to correctly classify examples. This paper investigates the robustness and resilience of various CNN architectures against black-box attack methods namely the Fast-Gradient Sign Method (FGSM) and Targeted Clean-Label Poisoning attacks. This research aims to identify what architectures demonstrate resistance to these attacks. Our findings reveal that simpler networks tend to exhibit more resilience compared to its more complex counterparts. The implications of these results are most likely more intricate than initially perceived, and this study merely scratches the surface of the world of neural network robustness. The insights provided through this study aim to provide guidance and provoke interest in the field in hopes to find a more practical application in the real world.

Introduction

With the recent advancements in the field of artificial intelligence, there is an escalating demand for a deeper understanding and comprehension of its capabilities and susceptibilities. This study delves specifically into the field of image/pattern recognition neural networks. As the utilization of these technologies extends upon more diverse fields and real world applications encompassing self-driving vehicles, spam detection, malware detection, stock trading to medical diagnostics, the need to examine the effects of adversarial perturbations, whether intentional or not, can compromise the accuracy of predictions and classifications made by these networks. The potential consequences of such attacks can be catastrophic from medical misdiagnoses to subverting fraud and malware detection algorithms where the integrity and accuracy of these algorithms is paramount. The main most common method of these attacks and what we will be examining in this study are black-box attacks, in which the assumption of the attacker has very limited information about the architecture and parameters of the network and is only able to access the network's output. This method of attack is much more applicable to real-life situations where attackers will have constrained insights into the inner workings of the target network.[1] Our study aims to gather insight on the intricacies, relevance and implications of black-box attacks.

Motivation and Objectives:

Our research endeavors aim to enhance our comprehension of various adversarial attack methods to understand the effects they have on image recognition neural networks. Firstly, we aim to explore and examine the effects of adversarial threats as well as explore and examine potential trends in vulnerabilities and further evaluate the effect of model architecture on resilience to adversarial attacks. In our pursuit, we have specifically opted for black-box attack methods as in the real-world, attackers will have limited knowledge and information about the target network. This scenario also includes unintentional attacks where the attacker might be unaware that the data they are manipulating is affecting a network. Our rationale for selecting this attack method is to mimic practical situations where the attacker's understanding of the system is constrained.

The two distinct attack methods have been selected as they represent potential real world situations. The Fast-Gradient Sign Method (FGSM) represents that of a deliberate attack, where the attacker requires some knowledge of the target system, accessing at least the network's output. This method utilizes the gradients produced by the model to perturb the images in order to maximize the loss, which aims to severely impact the networks ability to identify and classify images. Conversely, our second attack involves a Targeted Clean-Label poisoning attack meant to simulate the introduction of false information through incorrectly labeled images. This method imitates scenarios where users unknowingly upload data that may mislead the neural networks, such as uploading images with incorrect tags or search terms.

Methodology

To systematically investigate the impact of the aforementioned adversarial attack methods on image recognition neural networks, our methodology involved the selection of three distinct models: LeNet, AlexNet and ResNet20. These models were chosen due to their widespread use and varied architectures, guaranteeing a diverse evaluation of vulnerabilities. LeNet is the simplest network we chose consisting of 7 layers with 2 pooling and 3 convolution layers. AlexNet is the next step up in complexity with 11 layers with 3 pooling layers, and 5 convolution layers. Finally, we incorporated ResNet-20, a high-performing residual neural network capable of training tens to thousands of layers without sacrificing performance. ResNet is the closest to real life applications as it closely resembles modern AI systems such as GPT and ChatGPT as well as Tesla's HydraNet which utilizes various modified ResNet architectures to perform its self-driving tasks. Our rationale behind this diverse selection of models is to investigate the increasing complexities of neural networks and discern at what point adversarial attacks are most effective and why? Our approach aims to provide insights into the inherent vulnerabilities in different model architectures.

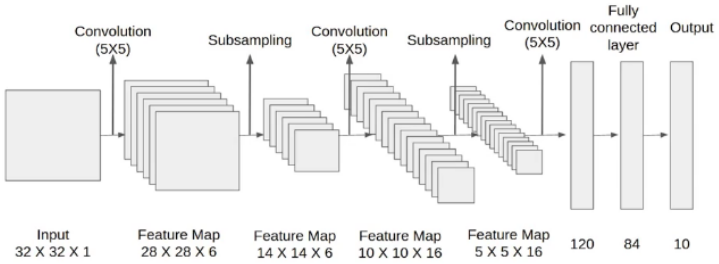
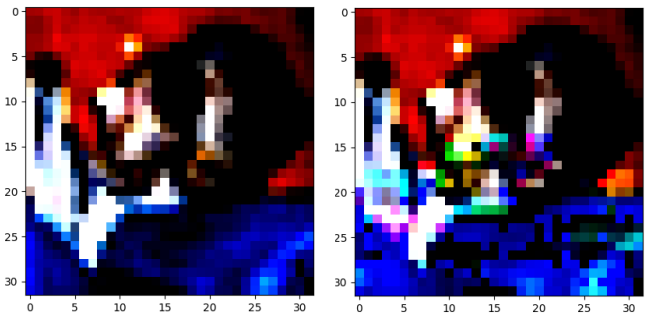


Figure 1. ***LeNet-5 model architecture.***

As for our attack methodology, we implemented the Fast-Gradient Sign Method (FGSM) as one of our primary attack methods. This attack method involves perturbing images to varying degrees through utilizing the gradient outputs from the outputs from image classification from each model. These gradients are then taken and applied to the model at varying degrees using epsilon in order to gauge the sensitivity of each model. We chose epsilon values of 0.01, 0.1, 0.25, 0.5, 0.75 and 1.0 in order to represent varying degrees of attacks. For each epsilon value, we record the resulting training and testing accuracies, loss and confidence levels.

Figure 2. FGSM Perturbations



Normal training image *Adversarial image (epsilon = 0.5)*

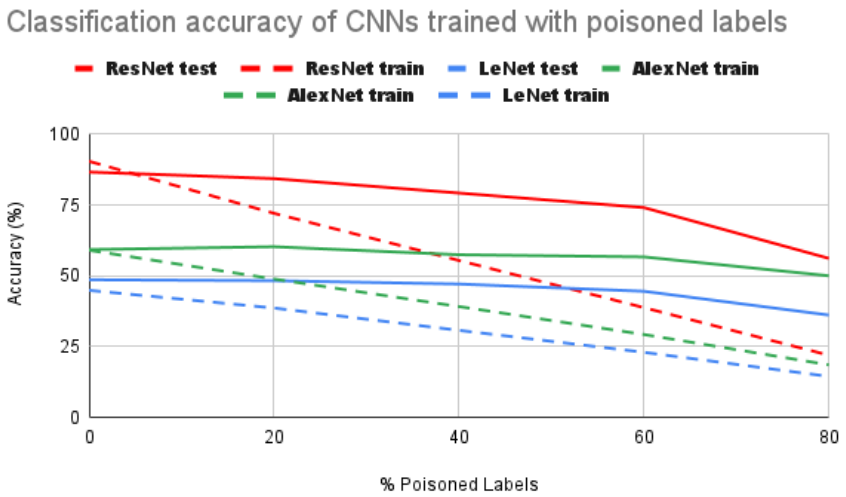
We also performed a Targeted Clean-Label attack to simulate the introduction of false information. This attack method involves poisoning a percentage of input labels in order to assess the models' robustness and resilience against this sort of label manipulation. We tested poisoning percentages of 0%, 20%, 40%, 60% and 80%. The training and test accuracies and loss were recorded for each model.

In order to ensure reproducibility, the experimental setup was standardized across all models, ensuring a fair and consistent evaluation. Each model was trained for 50 epochs with a 0.01 learn rate and a batch size of 128. While we wish we could have trained with more diverse parameters, both the time constraint and lack of processing power limited us in the scope of our experiments.

Results

As stated above, our three selected models were evaluated with the following hyperparameters: 50 epochs, 0.1 learn rate, 128 batch size

Figure 3. Model accuracies with varying degrees of adversarial training data



LeNet's testing accuracy with no poisoned labels was 48.610%. Its accuracy decreased as the proportion of poisoned labels used in training increased, culminating in a low of 36.160% when trained on 80% poisoned labels. The magnitude of the decreases started off small (a reduction of 0.330% when going from 0% poisoned to 20% poisoned)

but quickly ramped up to a decrease of 8.39% when going from 60% poisoned labels to 80%. Overall, this attack resulted in a maximum accuracy decrease of 12.450%. Table 1 further details LeNet's accuracy and loss across varying proportions of poisoned labels.

Table 1. *Lenet, 50 Epochs, 0.1 Learn Rate, 128 batch size*

% Poisoned Labels	Train Acc (%)	Train Loss	Test Acc (%)	Test Loss
0	44.880	1.576	48.610	1.511
20	38.640	1.854	48.280	1.542
40	30.764	2.060	47.110	1.682
60	23.016	2.195	44.550	1.846
80	14.500	2.284	36.160	2.104

With no poisoned labels, ResNet20 boasted a testing accuracy of 86.620%. ResNet20's accuracy decreased as the proportion of poisoned labels increased, ending off at a testing accuracy of 56.160%. ResNet20 experienced a trend similar to LeNet where the magnitude of the reductions increased as the proportion of adversarial training images increased (starting at a reduction of 2.320% and culminating in a final reduction of 17.92%). Overall, ResNet's accuracy decreased a total of 30.46%. Table 2 contains more comprehensive statistics regarding our label poisoning experiment on ResNet20.

Table 2. *Resnet20, 50 epochs, 0.1 Learn Rate, 128 batch size*

% Poisoned Labels	Train Acc (%)	Train Loss	Test Acc (%)	Test Loss
0	90.418	0.277	86.620	0.414
20	72.064	1.082	84.300	0.582
40	55.418	1.590	79.210	0.874
60	38.780	1.968	74.080	1.181
80	21.988	2.220	56.160	1.770

AlexNet started off with a testing accuracy of 59.270%, which dropped a total of 9.260%, ending off with 50.010% accuracy at 80% poisoned labels. Going from 0% poisoned to 20% did not appear to harm the model's predictive capabilities, resulting in an accuracy increase of 1.00%. AlexNet experienced very little accuracy reduction up to 60% poisoned labels, where it had an accuracy of 56.680%. However, moving up to 80% poisoned labels caused a much larger drop of 6.670%. Table 3 details our observations on how label poisoning affects AlexNet.

Table 3. *Alexnet, 50 epochs, 0.1 learn rate, 128 batch size*

% Poisoned Labels	Train Acc (%)	Train Loss	Test Acc (%)	Test Loss
0	59.030	1.270	59.270	1.274
20	48.866	1.682	60.270	1.289
40	39.160	1.937	57.450	1.415
60	29.296	2.119	56.680	1.625
80	18.594	2.257	50.010	1.927

The same hyperparameters were used for our FGSM tests for the sake of consistency. FGSM introduces a new hyperparameter, epsilon, which we set to the following values: 0.01, 0.1, 0.25, 0.50, 0.75, 1.00

LeNet's accuracy and confidence were heavily hindered by all values of epsilon tested. At the lowest epsilon value, 0.01, the model had an accuracy of 15.830% and matching confidence of 15.830%. As epsilon was increased, the model's predictive capabilities were continuously hindered, dropping as low as 11.700% at epsilon of 0.25. LeNet's performance did change significantly as epsilon was increased further, yielding an accuracy of 12.20% at an epsilon of 1.00. LeNet's classification confidence was inversely correlated with the drop in accuracy, increasing to a high of 39.959% at an epsilon value of 1.00.

Table 4. *LeNet, 50 Epochs, 0.1 Learn Rate, 128 batch size*

Epsilon	FGSM Acc (%)	FGSM Loss	FGSM Confidence
0.01	15.830	2.795	15.830
0.1	14.040	2.878	34.553
0.25	12.440	3.016	36.380
0.50	11.700	3.105	39.051
0.75	12.240	3.068	39.909
1.00	12.200	2.999	39.959

LeNet-5 exhibited a testing accuracy of 50.580%, loss of 1.467, and classification confidence of 52.198% under normal conditions.

ResNet20's highest accuracy was 34.930% at an epsilon value of 0.01. The model saw its largest drop in accuracy (~17% reduction) between moving from an epsilon value of 0.01 to 0.1. Aside from this, its accuracy dropped fairly steadily as the magnitude of the perturbations increased, ending at an accuracy of 10.550% at an epsilon value of 1.0. ResNet20's confidence started at 78.560% at an epsilon of 0.01 and dropped to 67.071% at an epsilon value of 1.0.

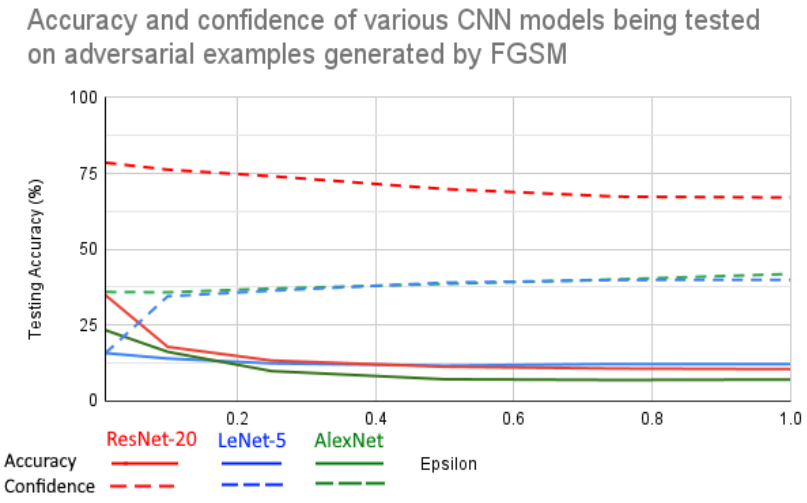


Figure 4. Accuracy and confidence intervals of various models under a FGSM attack.

Table 5. *Resnet20, 50 epochs, 0.1 Learn Rate, 128 batch size*

Epsilon	FGSM Acc (%)	FGSM Loss	FGSM Confidence
0.01	34.930	3.085	78.560
0.1	17.880	4.243	76.256
0.25	13.380	4.757	74.094
0.50	11.330	4.840	69.879
0.75	10.740	4.871	67.372
1.00	10.550	5.006	67.071

ResNet20 exhibited a testing accuracy of 85.120%, loss of 0.466, and classification confidence of 90.385% under normal conditions.

AlexNet's accuracy started at 23.380% and dropped to 7.110%. Its confidence started at 35.965% and increased to 41.909%, inversely proportional to its accuracy.

Table 6. *Alexnet, 50 epochs, 0.1 learn rate, 128 batch size*

Epsilon	FGSM Acc (%)	FGSM Loss	FGSM Confidence
0.01	23.380%	2.450	35.965%
0.1	16.240%	2.634	35.858%
0.25	9.900%	2.871	37.092%
0.50	7.210%	3.152	38.617%
0.75	6.970%	3.372	40.126%
1.00	7.110%	3.593	41.909%

AlexNet exhibited a testing accuracy of 59.010%, loss of 1.304, and a classification confidence of 58.339% under normal conditions.

COMPARATIVE ANALYSIS:

During our label poisoning experiments, AlexNet suffered the smallest reduction in accuracy. Additionally, its results stand out from ResNet and LeNet, as its predictive capabilities were not significantly impacted by adversarial training data up until 80% of training images were poisoned. On the other hand, ResNet and LeNet follow very similar trends where they suffer relatively steady drops in accuracy. However, it should be noted that ResNet yielded higher overall accuracies but faced larger reductions in accuracy when trained with adversarial inputs. ResNet's confidence decreased as perturbations increased. This is contrasted by LeNet, which exhibited higher confidence scores as perturbations increased.

Discussion

Our label poisoning results for accuracy tell an interesting story. ResNet-20 led the pack by a significant margin, followed by AlexNet, and finally LeNet. This mirrors the order of our models listed in descending order of depth as well as descending order of convolutional layers. At first glance it's easy to think that these results suggest that deeper models with more layers are more resilient against label poisoning, however the order of models above is also the descending list of models that experienced the largest reductions in accuracy. While it's possible that this suggests deeper models are more vulnerable to adversarial training attacks, we are hesitant to jump to said conclusion, as we cannot rule out that ResNet was disproportionately affected as the only model that can classify images with a high degree of accuracy (85% under normal conditions, followed by AlexNet's 59% accuracy). In other words, it is possible AlexNet and LeNet did not experience a similar level of classification disruption due to the fact that they were not very good at the task of recognizing features in the first place. We would like to research this further, especially using more deep CNNs with vastly different architectures.

Additionally, we observed that despite having the least convolutional layers (as well as least layers overall), LeNet-5 was the model that had its accuracy affected the least by large perturbations (epsilon values of 0.5<). This suggests that model depth (number of layers) is a meaningless metric on its own when it comes to robustness against adversarial attacks. This is further supported by the results yielded by ResNet20. ResNet20 proved itself to be the most capable model under normal circumstances, however it was the most susceptible to both label poisoning and FGSM attacks. All models had similar reductions in confidence when switching from real examples to adversarial examples. However, AlexNet and LeNet would classify the adversarial examples with low confidence, while ResNet's confidence ranged from 67-78% . Despite boasting the highest accuracy of all tested models, it is troubling that its classification confidence remains disproportionately higher than its accuracy while undergoing FGSM attacks.

Conclusion

In our analysis, we explored two black-box adversarial attack methods that target neural networks at different stages: the Fast-Gradient Sign Method (FGSM) for production-phase attacks and targeted clean-label poisoning for training phase. These attacks are relevant in the era of growing artificial intelligence and user-sourced data, and have potential implications. The FGSM attack effectively causes neural networks to misclassify target examples whilst remaining undetectable, even by the human eye. Similarly, the label poisoning attack severely compromises a network's ability to classify unaltered examples. This study aims to highlight the risks of training neural networks using unfiltered and easily manipulable data from the internet, suggesting the need for cautious consideration and security measures and further investigation into methods of making neural networks more robust and less susceptible to these types of attacks.

References

[1] Analytics Vidhya, Screenshot-from-2021-03-18-12-47-59, 2023. Accessed: Dec 8, 2023. [Photo]. Available: https://www.analyticsvidhya.com/blog/2021/03/the-architecture-of-lenet-5/

[2] N. Narodytska and S. Kasiviswanathan, "Simple Black-Box Adversarial Attacks on Deep Neural Networks," in 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, 2017, pp. 1310-1318, doi: 10.1109/CVPRW.2017.172.

[3] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards Deep Learning Models Resistant to Adversarial Attacks," 2017, doi: 10.48550/arXiv.1706.06083.

[4] I. Goodfellow and J. Shlens, "Explaining and Harnessing Adversarial Examples," [Online]. Available: https://www.researchgate.net/publication/269935591_Explaining_and_Harnessing_Adversarial_Examples. [Accessed: Dec. 8, 2023].

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," [Online]. Available: https://arxiv.org/abs/1512.03385. [Accessed: Dec. 8, 2023].

[6] S. T. Krishna and H. K. Kalluri, "Deep learning and transfer learning approaches for Image Classification," [Online]. Available: https://www.researchgate.net/profile/Hemantha-Kumar-Kalluri/publication/333666150_Deep_Learning_and_Transfer_Learning_Approaches_for_Image_Classification/links/5cfbeeb9a6fdccd1308d6aae/Deep-Learning-and-Transfer-Learning-Approaches-for-Image-Classification.pdf. [Accessed: Dec. 8, 2023].

[7] D. Ciresan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, "Flexible, High Performance Convolutional Neural Networks for Image Classification," in International Joint Conference on Artificial Intelligence (IJCAI-2011), 2011, pp. 1237-1242, doi: 10.5591/978-1-57735-516-8/IJCAI11-210.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional Neural Networks," Communications of the ACM, vol. 60, no. 6, pp. 84–90, 2017, doi: 10.1145/3065386.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in Computer Vision – ECCV 2016, 2016, pp. 630–645, Sep. 2016, doi: 10.1007/978-3-319-46493-0_38.

[10] S. Shan et al., "Gotta catch'em all: Using honeypots to catch adversarial attacks on Neural Networks," in Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security, Apr. 2020, doi: 10.1145/3372297.3417231.

[11] N. Vemuri, "Scoring Confidence in Neural Networks," Master's thesis, EECS Department, University of California, Berkeley, Jun. 2020. [Online]. Available: http://www2.eecs.berkeley.edu/Pubs/TechRpts/2020/EECS-2020-132.html. [Accessed: December 8th, 2023].

[12] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: A Simple and Accurate Method to Fool Deep Neural Networks," [Online]. Available: https://arxiv.org/abs/1511.04599. [Accessed: Dec. 8, 2023].

[13] GMT710 and BigBallon, "CIFAR-ZOO: A Collection of Pre-trained Models for CIFAR-10/100," Version 1.0. [Online]. Available: https://github.com/BIGBALLON/CIFAR-ZOO. [Accessed: Month Day, Year].