Deceptive Pixels: Exploring the Landscape of Visualization Attacks in Neural Networks

Mohamed A. Abdel Hamid*
Department of Computer and Information Sciences
University of St. Thomas, St. Paul
2115 Summit Ave, St Paul, MN 55105
abde8473@stthomas.edu

***Abstract*— Adversarial model manipulation and adversarial training have emerged as significant areas of study in the context of enhancing the robustness of machine learning models against adversarial attacks. This study focuses on the exploration and analysis of these techniques, employing PyTorch for the implementation and evaluation of various adversarial attacks. The research builds upon and leverages two types of adversarial attacks: Passive and Active fooling, each with its unique approach to misleading model interpretations. Unlike traditional adversarial attacks that perturb input data, this attack fine-tunes a pre-trained model to generate misleading interpretations while maintaining the model's original performance. Passive fooling, including Location Fooling, Top-k Fooling, and Center-mass Fooling, generate uninformative interpretations while maintaining the model's accuracy. In contrast, Active fooling manipulates the model to produce completely incorrect interpretations that switch explanations between target classes.**

Keywords—Adversarial Model Manipulation, Neural Network, Adversarial examples, Adversarial Training, Machine Learning, Pytorch, Resnet50.

I. Introduction

Adversarial Model Manipulation (AMM) is a rather intricate and devious class of adversarial attacks that disrupt machine learning model interpretability by discreetly modifying their parameters. Unlike traditional adversarial attacks, which primarily focuses on perturbing input data, AMM operates invisibly, modifying an existing model to engender misleading interpretations while seemingly keeping the model's original performance measures. This study delves into an examination and analysis of AMM and the related adversarial training approaches, utilizing the PyTorch framework for adversarial attack instantiation and assessment. Our research divides the terrain of adversarial strategies into two major categories: passive fooling and active fooling. Passive Fooling is a type of adversarial approach that generates obscured interpretations while possibly retaining the predicted accuracy of the model. Location Fooling, Top-k Fooling, and Center-mass Fooling are all belonging to this category. Location Fooling tricks the model into producing interpretations that highlight unimportant spatial locations within the picture. Top-k Fooling, on the other hand, draws attention to the image's non-informative top-k pixels. While, Center-mass Fooling works on the interpretative perplexity by moving the interpretation's center of mass.

This research was hosted by Old Dominion University and was sponsored by NSF (National Science Foundation).

Active Fooling, on the other hand, is a more active type of approach. It manipulates the model to provide switch interpretations between classes, when it misdirects attention to unimportant items in the image, resulting in fundamentally faulty conclusions. Interpretability approaches such as Layer-wise Relevance Propagation (LRP), Grad-CAM, and Simple-Grad are used to analyze the decision-making process of neural networks. These strategies provide detailed insights into the model's internal processes and highlight areas of interest within the input space. Resnet50, a 50-layer deep convolutional neural network known for its excellency in picture classification and other vision-centric tasks, is used in our experimental framework. This model's resilience and complexity make it a suitable testbed for evaluating the effectiveness in different adversarial attacks. The goals of this study are multidimensional. We want to further understand the subtle variations between passive and aggressive deception methods and how they affect model interpretability. We are also attempting to measure the effectiveness of various adversarial assaults on the model's performance and interpretability. Finally, and in the future, we want to provide viable solutions to protect against adversarial assaults through this extensive investigation.

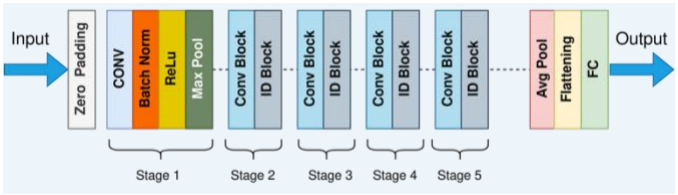

Figure 1: Resnet-50 model architecture

II. Unveiling Adversarial Model Manipulation

The innovative notion of Adversarial Model Manipulation is explored in this section. This nuanced type of attack can is a tactic that modifies model parameters to fool interpretation algorithms. Unlike traditional adversarial assaults, which alter input data, this strategy quietly adjusts an existing model to provide false interpretations. Surprisingly, the modified model retains the clean model's performance during training. Adversarial training can be used to increase the

resilience of the ResNet50 model in the setting of visual attacks. This entails creating adversarial instances, which are stimuli meant to induce the model to make a mistake, and then training the model on these examples to improve its resistance to future attacks of this sort. The model used for the Resnet50 dataset has a 100% accuracy. The process of adversarial training goes as follows. First, the model must be trained on batches of data photos. The size of the batch used in the Resnet50 model consists of 64. After then, the training began with 50 epochs. Each epoch consisted of 256 iterations. The dataset consists of 4 different training folders, for Top-k, Center-mass, Location fooling and Clean images. Folders each consist of 900 images compressed into a .npz file format ready to go for the classifier to be trained. Testing folders alike, and they consist of 100 images each.

III. The Fooling Polarity: Passive and Active

The landscape of adversarial attacks is bifurcated into two primary categories: Passive Fooling and Active Fooling. Each category represents a unique approach to manipulating the interpretability of machine learning models, with distinct objectives and methodologies.

A. Passive Fooling

Passive Fooling is a type of adversarial approach in which the model is meticulously manipulated to change its interpretations, highlighting uninformative parts of each image. This method successfully obscures the most important facts while emphasizing less crucial or wholly irrelevant aspects of the image. This study examines three forms of Passive Fooling: Location Fooling, Top-k Fooling, and Center-mass Fooling. The purposeful change of explanations to emphasize the uninformative peripheries of each image, even when the item of interest is centrally positioned, is known as location fooling. Top-k Fooling, on the other hand, drastically affects the most emphasized top-k pixels after fooling, modifying the interpretation's focus. Center-mass Fooling goes this a step further by changing the heatmaps' centers to meaningless portions of the pictures, resulting in confusion.

B. Active Fooling

In contrast according to Juyeon Heo et al. (2019), Active Fooling entails a more vigorous modification of the model's interpretability. The descriptions for two target classes are swapped in this technique. The research uses two classes of interest, c1 and c2, and a dataset that contains items from both classes in each image to demonstrate this notion. In this situation, the penalty term is intended to shift the explanation for c1 to that of c2, and vice versa. While doing the Active Fooling, this strategic manipulation preserves categorization accuracy. The authors discover that in VGG19 and virtually in ResNet50, but not in DenseNet121, the explanations for c1 and c2 are clearly switched, implying a relationship between model complexity and the degree of Active Fooling.

While both Passive and Active Fooling aim to alter the interpretations without significantly compromising the accuracy of the original models, they differ in their focus. Passive Fooling emphasizes uninformative parts of the image, effectively hiding the most salient evidence. Active Fooling, on the other hand, involves swapping the explanations for two target classes. These distinct approaches underscore the complexity and diversity of adversarial attacks in the realm of machine learning.

IV. The Art of Fooling

A machine learning model is used to process a picture, which can be a digital representation of any item, person, or scene, in the context in question. This model has filters, which are tiny matrices that change the input data (in this example, the picture) into a form that the model can learn from.

The idea of a 'gradient' comes into play in this approach, which in the context of machine learning refers to the slope of a function at a certain location. The gradient essentially serves as a guide for us to alter our parameters (filters) to improve the model's performance.

When 'ranking filters,' the process begins by sorting these filters in a certain order, frequently depending on their contribution to the model's final output. These filters are then modified in '10 increments,' implying that the filter values are altered incrementally to track the subsequent changes in the model's output. Grad-CAM (Gradient-weighted Class Activation Mapping) is a method used to depict the model's attention or the regions in the picture that it thinks most relevant. An image's 'initial Grad-CAM' refers to the Grad-CAM created before to any filter alterations and serves as a baseline to compare with future Grad-CAMs generated after each filter modification.

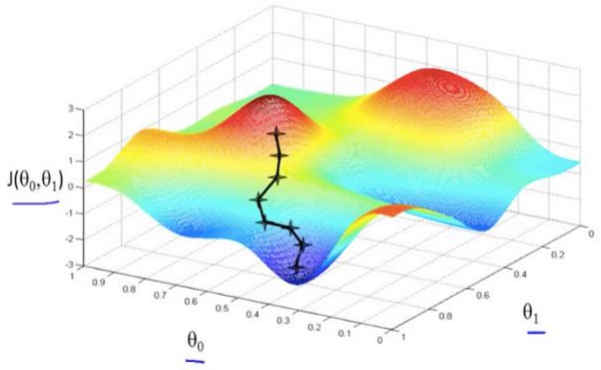

Figure 2: Gradient Calculation

V. Methodology

A. Hardware

All the tests in the research were performed on a 2020 M1 Mac Machine with an M1 chip, featuring an 8-core CPU and 8-core GPU. The machine also had 16GB of unified memory and ran on macOS Ventura, the latest operating system at the time of the experiments. The M1 chip, known for its power efficiency and performance, provided a solid foundation for conducting the research experiments effectively and efficiently.

B. Google Colab

Google Colab is an open-source, interactive web tool used to run code and make annotations about it and allows you to run and save the notebook on your Google Drive, facilitating easy access and sharing with others.

C. PyTorch

This platform facilitates the development of machine learning algorithms by making it possible for participants to train models and dynamic computational graphs.

VI. Results

Our investigation produced compelling results, demonstrating the effects of training classifiers in a combination of Top-l or Center-mass or Location fooling attacks and clean data. Despite the integration of these fooling attacks, the classifiers exhibited 100% training accuracy, indicating the robustness and resilience of our models. Next step would be extensively testing the trained classifier with validation dataset.

Fooling Type	Training Classifier Accuracy
Top-k	100%
Center-mass	100%
Location	100%

Figure 3: Value of Accuracy

VII. Discussion

The study reveals that Passive fooling, which includes Location Fooling, Top-k Fooling, and Center-mass Fooling, can generate uninformative interpretations while maintaining the model's accuracy. This implies that the model, while appearing to perform well, is focusing on irrelevant locations or pixels within the image, eventually leading to potentially misleading interpretations.

On the other hand, Active fooling manipulates the model to produce completely incorrect interpretations that switch explanations between target classes. Despite the misleading interpretations, the model's accuracy remains intact, demonstrating the deceptive nature of these attacks. These attacks are particularly effective against interpretation methods such as Layer-wise Relevance Propagation (LRP), Grad-CAM, and Simple-Grad, which are typically employed to understand the decision-making process of neural networks.

VIII. Conclusion

The vulnerability of these models to hostile instances emphasizes the need for adversarial training in the domain of machine learning. Machine learning models, despite their excellent accuracy under normal settings, can be readily duped by hostile cases. These events, which are frequently unnoticeable to the untrained human eye from ordinary inputs, might cause the model to make inaccurate predictions. Adversarial training is an effective countermeasure that fortifies models against certain erroneous attacks.

Adversarial training involves supplementing the original training data with adversarial cases. We strengthen the model's resilience against adversarial assaults by exposing it to these false situations throughout the training process.

However, the path to effective adversarial training is fraught with challenges, the most prominent of which is the generation of efficacious adversarial examples. These instances must be meticulously crafted to deceive the model. Striking this delicate balance is a non-trivial task and represents a pivotal area for future research.

In conclusion, our exploration of adversarial model manipulation and adversarial training methodologies has underscored the importance of these techniques in enhancing the robustness of machine learning models. The nuanced understanding of passive and active fooling tactics, coupled with the evaluation of the resilience of models against different adversarial attacks, paves the way for the development of more robust and reliable machine learning systems. Future work should be focused on overcoming the challenges in adversarial training and further refining our understanding of the vulnerabilities and defenses of machine learning models.

IX. Acknowledgment

This research experience was sponsored by the National Science Foundation and hosted by The Old Dominion University. Special thanks to my mentor Dr. Jiang Li. Special thanks to the directors of the program Dr. Chunsheng Xin, Dr. Peng Jian and Dr. Khan Iftekharuddin for the opportunity to participate in the research program. Special thanks to PhD student (Shahab Uddin) that guided me throughout this research.

REFERENCES

[1] Juyeon Heo, Sunghwan Joo, Taesup Moon, 2019. Fooling Neural Network Interpretations via Adversarial Model Manipulation. https://arxiv.org/abs/1902.02041

[2] Suvaditya Mukherjee, 2022. The Annotated ResNet-50, Explaining how ResNet-50 works and why it is so popular. Published in Towards Data Science. https://towardsdatascience.com/the-annotated-resnet-50-a6c536034758

[3] Renu Khandelwal, 2018. Machine learning : Gradient Descent. Published in Medium. https://arshren.medium.com/gradient-descent-5a13f385d403

[4] Google Colab. https://colab.research.google.com

[5] Pytorch. https://pytorch.org

[6] Grad-CAM Reveals the Why Behind Deep Learning Decisions. https://www.mathworks.com/help/deeplearning/ug/gradcam-explains-why.html