Deep Learning Project III
Jail Breaking Deep Models

Team GradeIsAllYouNeed

Ansh Sarkar[1], Princy Doshi[2], Simran Kucheria[3]

[1]as20363@nyu.edu, [2]pd2672@nyu.edu, [3]sk11645@nyu.edu

New York University

Abstract

This project aims to understand the impact of various jail-breaking (adversarial attack) techniques on making the performance of a deep model worse. Starting with a pretrained ResNet-34, we evaluated multiple attack methods-including iterative and targeted attacks-to identify the most effective approach. All techniques were then applied to DenseNet-121 to compare model robustness. The MI-FGSM technique performed the best on ResNet with a top1 accuracy of 0% and transferred fairly well to DenseNet, giving us our lowest DenseNet top1 accuracy of 45%

Implementation can be found here: https://github.com/anshsarkar/ECE-GY-7123-Deep-Learning-Project-3

Overview

Jailbreaking deep models refers to the process of launching adversarial attacks on production-grade, publicly available models to significantly degrade their performance.

In this project, the focus is on image classifiers, where the attacks are engineered by creating adversarial samples of the input by designing attacks where the perturbed test image remains visually similar to the original.

We keep this difference in check by adding limits to the L_0 (which upper-bounds how many pixels are allowed to be perturbed) and L_∞ (which upper-bounds how much each image pixel is allowed to be perturbed).

The project achieved the following objectives:

- Baseline accuracy of a pretrained ResNet model on a sample dataset.
- Applying pixel-wise attacks and observing their effects on the same dataset. The Fast Gradient Sign Method (FGSM) was applied, and its impact on classification performance was analyzed.
- Experimenting with other techniques to improve the degradation, including gradient-based adversarial attacks, targeted attacks, or attacks using optimizers.
- Exploring the effect of patch attacks. Analyze changes in classification accuracy and compare their effectiveness to full-image adversarial attacks.
- Transferability to other models. By examining how adversarial examples affect multiple models, we aim to assess the generalizability of attacks and highlight potential risks to black-box systems in real-world applications.

Methodology

Our methodology involved first establishing a baseline, followed by implementing Fast Gradient Sign Method (FGSM) - a pixel-wise attack. This was followed by experiments involving other pixel-wise attacks wherein we switched out the optimizers and also used a multi-gradients approach to understand their effects.
We also experimented with the patched version of the MI-FGSM attack before finally concluding our experiments by applying all techniques to a DenseNet model to understand its effectiveness.

To get better insights into the model performance, top-k accuracy metrics were also calculated for k = 1 and k = 5, where top-k computes the k most likely class labels according to the classifier, and if the correct label is within those, we accept it as a valid prediction.

To ensure that the adversarial examples stay within the allowed perturbation limit ($\epsilon = 0.02$), we computed the L_∞ distance between each original image x and its adversarial counterpart x'. The L_∞ norm measures the largest absolute pixel-wise difference, ensuring that no pixel in the adversarial image differs from the original by more than ϵ.

The L_∞ distance between the original image x and its adversarial counterpart x' is defined as:

$$L_\infty(x, x') = \max_i |x_i - x'_i|$$

where x_i and x'_i represent the pixel values at position i.
We also wrote a function to verify that our L_∞ is within the ϵ bounds.

Baseline

To perform experiments, we first established a baseline on the test dataset using a pretrained ResNet34 that was trained on the ImageNet dataset.
To evaluate the performance, we first standardised and normalised all the images in the testing dataset before passing them on to the models.
The baseline is tabulated below

Baseline	
Accuracy Metric	**Accuracy**
Top-1	76%
Top-5	94.2%

Table 1: Baseline Model performance for ResNet34

Pixel Wise Attacks

Pixel-wise attacks focus on the L_∞ distance. We performed an attack using the Fast Gradient Signing Method.

FGSM distorts the image by implementing a single step of gradient ascent (in pixel space) and truncating the values of the gradients to at most ϵ.

$$x \leftarrow x + \epsilon \cdot \text{sign}(\nabla_x L)$$

- x: The input to the model (e.g., an image or text).
- L: The loss function, typically cross-entropy loss.
- $\nabla_x L$: The gradient of the loss L with respect to the input x.
- sign$(\cdot)$: The sign function, which projects the gradient onto the unit L_∞ cube.
- ϵ: A small positive scalar controlling the magnitude of the perturbation.

We implemented the FGSM attack on the test dataset using the torchattacks library available in PyTorch. The value of ϵ was capped at 0.02. The function takes the normalised image as input and gives a normalised adversarial image as input. We denormalised and saved all these in a new dataset called AdversarialTestSet1.
The results are tabulated below

FGSM		
Accuracy Metric	**Accuracy**	**Relative Accuracy Drop**
Top-1	5.8%	70.2%
Top-5	31.2%	73%

Table 2: FGSM Model performance for ϵ = 0.02

The perturbations caused a drastic drop in accuracy just by changing the pixels. The resultant accuracies for top-1 and top-5 dropped by 70% relatively.

Image comparisons between the normalised images and their normalised adversarial counterparts do not show much difference visually, but there is a major difference in the predictions of the classifier on the adversarial test.

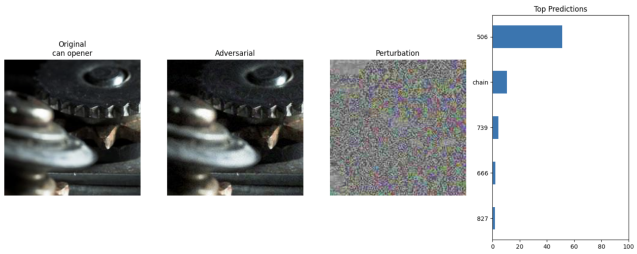

Figure 1: Image Comparisons for FGSM with prediction confidences

Improved Attacks

After implementing FGSM, we shifted our focus to further degrade the performance of the ResNet-34 image classifier whilst keeping the ϵ constant. We enhanced the FGSM attack by increasing the number of gradient steps(BIM), incorporating projected gradient descent(PGD), which is an iterative attack that applies multiple small gradient steps, and also experimented with a momentum-based variant of the optimizer(MI-FGSM). Other approaches like MI-FGSM with cosine Annealing and an Adam-based PGD were also tried, but the results did not have significant improvements over MI-FGSM and hence are not being considered here. Those results can be found on our GitHub repo.

BIM The Basic Iterative Method (BIM) enhances the standard FGSM by applying multiple small, iterative perturbations, each constrained within the L_∞ norm bound ($\epsilon = 0.02$). At each step, the adversarial image is clipped to ensure that the perturbation remains imperceptible and within the allowed pixel value range. We experimented with 10 iterations.

PGD Projected Gradient Descent (PGD) is an iterative adversarial attack technique that generates adversarial examples by repeatedly applying small, calculated perturbations to an input in the direction that increases the model's loss, all while ensuring that the perturbed input remains within a specified distance—typically an L_∞ ball of radius ϵ—from the original input. After each update, the adversarial example is projected back into this allowed region to maintain the perturbation constraint, resulting in a more effective and reliable attack compared to single-step methods like FGSM. The α = 0.004 and the number of iterations = 15 gave us the best PGD results but they were not as good as MI-FGSM results

MI-FGSM In Momentum-Interative FGSM, we used the momentum optimiser on top of an iterative FGSM, which allows the attack to accumulate gradient information across steps and escape poor local maxima. With this, it generates stronger adversarial perturbations within the same constraint. The values of α = 0.002 and the number of iterations = 10 with a momentum value of 0.9 gave us the best results. The dataset generated out of this method was saved as AdversarialTestSet2.

We also experimented with different iterations and α values for MI-FGSM to observe the classification accuracy drop.

Variations			
Method	**Accuracy Metric**	**Accuracy**	**Relative Accuracy Drop**
BIM	Top-1	1.6%	74.2%
BIM	Top-5	42.6%	51.8%
PGD	Top-1	0.2%	76%
PGD	Top-5	24.6%	66.74%
MI-FGSM	Top-1	0%	76%
MI-FGSM	Top-5	10.80%	80.6%

Table 3: Model performance for $\epsilon = 0.02$

Variations on MI-FGSM			
α Values	**Iterations**	**Accuracy - Top1**	**Accuracy - Top5**
0.002	20	0.00%	14.20%
0.002	10	0.00%	10.80%
0.002	30	0.40%	17.20%
0.2	20	0.20%	18.80%
0.2	10	0.20%	16.00%
0.2	30	1.20%	23.20%

Table 4: Model performance when varying iterations and α

Evaluating the above models on this dataset demonstrated a substantial drop in top-1 and top-5 accuracy, highlighting the increased potency of combining a momentum optimizer with an iterative FGSM approach.
Based on the improved results of MI-FGSM, we decided to implement it in a patched format.

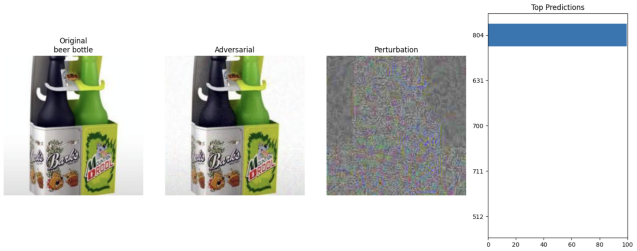

Figure 2: Image Comparisons for MI-FGSM with prediction confidences

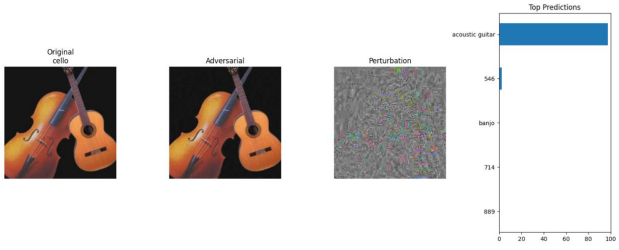

Figure 3: Image Comparisons for BIM with prediction confidences

Patch Attacks

For the patch-based adversarial attack, we employed the Momentum Iterative Fast Gradient Sign Method (MI-FGSM) to perturb only a small, randomly selected 32×32 region within each test image. We also changed the implementation to a targeted attack. Since this was a patched attack, we also changed the ϵ values to be 0.5.
To further improve the performance, the values of α were set to 0.002, and the number of iterations = 10 with a momentum value of 0.9.

The results are tabulated below

Patched MI-FGSM		
Accuracy Metric	**Accuracy**	**Relative Accuracy Drop**
Top-1	16.60%	78.16%
Top-5	60.60%	35.67%

Table 5: Patched MI-FGSM Model performance for $\epsilon = 0.02$

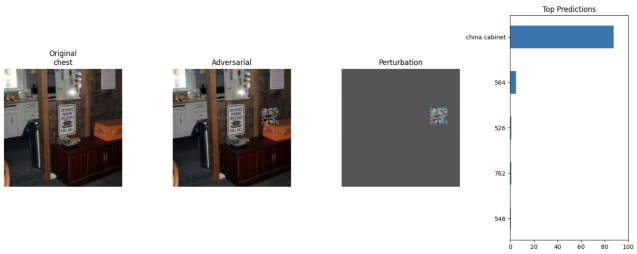

Figure 4: Image Comparisons for Patched MI-FGSM with prediction confidences

The dataset generated out of this method was saved as AdversarialTestSet3.

DenseNet Transferability

After experimenting with different adversarial attack methods and generating corresponding adversarial datasets primarily on architectures like ResNet, we aimed to evaluate how well these adversarial examples transfer to a different model, specifically a pretrained DenseNet.

The process began by establishing a baseline accuracy for DenseNet on the dataset, which served as a reference for its standard performance. Subsequently, each adversarial dataset, crafted using various attack strategies and originally targeting different models, was systematically applied to the DenseNet to assess its robustness.

Baseline	
Accuracy Metric	**Accuracy**
Top-1	74.8%
Top-5	93.6%

Table 6: Baseline DenseNet performance

The attacks, while not performing like they do on ResNet, did show good drops in accuracy, which indicates that the

DenseNet			
Dataset	**Accuracy Metric**	**Accuracy**	**Relative Accuracy Drop**
Adversarial Dataset 1 (FGSM)	Top-1	48.8%	26%
Adversarial Dataset 1 (FGSM)	Top-5	77.4%	16.2%
Adversarial Dataset 2 (MI-FGSM)	Top-1	45.0%	27.4%
Adversarial Dataset 2 (MI-FGSM)	Top-5	79.4%	13%
Adversarial Dataset 3(Patch)	Top-1	67.0%	2%
Adversarial Dataset 3(Patch)	Top-5	90.2%	1.2%

Table 7: Other Dataset performance

adversarial images did generalise well to obfuscate the input to other models. The best-performing attack of MI-FGSM was the best-performing attack on DenseNet as well.
The results also show us how even with just minor pixel differences the accuracy of the entire model can be brought down.
The patched attack doesnt perform as well on densenet as it does on resnet - this could be due to the patched images not generalising well enough on densenet given the small area under consideration.
To mitigate the effects of adversarial images on models we could train them on data that includes these adversarial images. We could also use an ensemble of models to make models more robust to adversarial images, as one model might underperform, but the others would likely not be influenced by the adversarial images. We could also implement another model that detects anomalous images and reconstructs it in such a way that it doesnt interfere with the model outputs.

Results

Among all the attack variations tested on the ResNet architecture, the Momentum Iterative Fast Gradient Sign Method (MI-FGSM) consistently demonstrated superior effectiveness.

This was observed through a significant reduction in classification accuracy when adversarial examples generated by MI-FGSM were applied to the ResNet-34 model. The method's use of momentum not only enhances attack strength in white-box settings but also significantly boosts the transferability of adversarial examples to other models, making it effective in black-box scenarios as well.

The results tables also outline how different methods perform when applied to a DenseNet model, showing comparable results hence demonstarting good transferability.

DenseNet			
Dataset	**Model**	**Accuracy Metric - Top-1**	**Accuracy Metric - Top-k**
Baseline	ResNet	76%	94.2%
Baseline	DenseNet	74.3%	93.6%
Adversarial Dataset 1 (FGSM)	ResNet	5.8%	31.2%
Adversarial Dataset 1 (FGSM)	DenseNet	48.8%	77.4%
Adversarial Dataset 2 (MI-FGSM)	ResNet	0.0%	10.8%
Adversarial Dataset 2 (MI-FGSM)	DenseNet	45.0%	79.4%
Adversarial Dataset 3 (Patch)	ResNet	16.6%	60.6%
Adversarial Dataset 3 (Patch)	DenseNet	67.0%	90.2%

Table 8: Performance of DenseNet and ResNet models on various datasets, including adversarial datasets.

References

Dong, Y.; Liao, F.; Pang, T.; Su, H.; Zhu, J.; Hu, X.; and Li, J. 2018. Boosting Adversarial Attacks with Momentum. *arXiv preprint arXiv:1710.06081*.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2015. Explaining and Harnessing Adversarial Examples. arXiv:1412.6572.

Kim, H. 2018a. FGSM-pytorch: A PyTorch Implementation of "Explaining and Harnessing Adversarial Examples". https://github.com/Harry24k/FGSM-pytorch. Accessed: 2025-05-13.

Kim, H. 2018b. PGD-pytorch: A PyTorch Implementation of "Towards Deep Learning Models Resistant to Adversarial Attacks". https://github.com/Harry24k/PGD-pytorch. Accessed: 2025-05-13.

Kim, H. 2020. Torchattacks: A pytorch repository for adversarial attacks. *arXiv preprint arXiv:2010.01950*.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2018. Towards Deep Learning Models Resistant to Adversarial Attacks. *arXiv preprint arXiv:1706.06083*.

Sen, J.; Sen, A.; and Chatterjee, A. 2023. Models: Analysis and Defense. ArXiv preprint arXiv:2312.16880.

Waghela, H.; Sen, J.; and Rakshit, S. 2024. Robust Image Classification: Defensive Strategies against FGSM and PGD Adversarial Attacks. arXiv:2408.13274.