**Tables 1, 2, 2\_1 are before adversarial training. Tables 4, 5, 5\_1 are after adversarial training. All experiments are on perturb=0.03**

**Table 1.** Clean Performance of MLP alternatives ensemble modules for SEVIT (VIT + Ensemble modules) for Majority Voting

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ensemble Model** | **m = 1** | **m = 2** | **m = 3** | **m = 4** | **m = 5** |
| MLP (625.22M) | 94.203% | 96.522% | 95.362% | 95.797% | 95.797% |
| CNN (1.03M) | 93.043% | 96.232% | 96.232% | 94.058% | 94.058% |
| ResNet-FT (13.59M) | 95.362% | 93.913% | 93.768% | 90.580% | 90.580% |
| ResNet-TL (2.41M) | 93.043% | 92.464% | 93.333% | 91.014% | 92.029% |
| ResNet-FT-CNN (14.27M) | 96.377% | 92.754% | 93.188% | 85.797% | 85.797% |
| ResNet-TL-CNN (3.09M) | 93.768% | 93.478% | 94.203% | 92.029% | 92.029% |

**Table 2.** Adversarial Attack Performance of MLP alternatives ensemble modules for SEVIT (VIT + Ensemble modules) via Majority Voting for m=3. Here FGSM, PGD, BIM, and AutoPGD attack samples are having perturbation budget equal to 0.03

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Ensemble Model** | **Clean Samples** | **FGSM** | **PGD** | **BIM** | **AutoPGD** | **C&W** |
| ViT | 96.377% | 55.652% | 32.323% | 28.994% | 23.768% | 47.836% |
| MLP | 95.362% | 84.638% | 82.754% | 95.362% | 81.304% | 95.362% |
| CNN | 96.232% | 87.391% | 85.217% | 96.232% | 84.928% | 96.232% |
| ResNet-FT | 93.768% | 85.507% | 82.174% | 93.768% | 80.580% | 93.768% |
| ResNet-TL | 93.333% | 81.884% | 76.522% | 93.333% | 75.652% | 93.333% |
| ResNet-FT-CNN | 93.188% | 80.870% | 76.957% | 93.188% | 74.638% | 93.188% |
| ResNet-TL-CNN | 94.203% | 84.928% | 82.174% | 94.203% | 80.725% | 94.203% |

**Table 2\_1.** Adversarial Attack performance of surrogate and SEVIT models (with MLP alternatives) on attack samples via Majority Voting, where these attack samples are generated on the surrogate model through model extraction attack

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Ensemble Model** | **Clean Samples** | **FGSM** | **PGD** | **BIM** | **AutoPGD** | **C&W** |
| CNN Surrogate | **90.14%** | **90.72%** | **90.58%** | **91.45%** | **90.58%** | **91.45%** |
| MLP | 95.362% | 92.899% | 93.188% | 95.362% | 94.348% | 95.362% |
| CNN | 96.232% | 94.348% | 94.783% | 96.232% | 95.362% | 96.232% |
| ResNet-FT | 93.768% | 94.783% | 94.928% | 93.768% | 94.783% | 93.768% |
| ResNet-TL | 93.333% | 92.029% | 92.174% | 93.333% | 92.029% | 93.333% |
| ResNet-FT-CNN | 93.188% | 94.203% | 94.203% | 93.188% | 94.348% | 93.188% |
| ResNet-TL-CNN | 94.203% | 92.174% | 92.319% | 94.203% | 92.464% | 94.203% |

**------------------------- AFTER ADVERSARIAL TRAINING-------------------------------**

**Table 4.** Clean Performance of MLP alternatives ensemble modules for SEVIT (VIT + Ensemble modules) after adversarial training. Here abbreviations are as, MW: Majority Voting, A: Averaging, and WA: Weighted Averaging

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ensemble Model** | **m = 1** | **m = 2** | **m = 3** | **m = 4** | **m = 5** |
| MLP (625.22M) | 91.449% | 95.507% | 95.652% | 95.507% | 95.507% |
| CNN (1.03M) | 95.507% | 95.652% | 96.087% | 94.203% | 94.348% |
| ResNet-FT (13.59M) | 96.087% | 96.377% | 96.232% | 91.449% | 91.449% |
| ResNet-TL (2.41M) | 91.594% | 90.290% | 93.333% | 91.449% | 91.594% |
| ResNet-FT-CNN (14.27M) | 95.942% | 95.942% | 96.522% | 88.986% | 89.130% |
| ResNet-TL-CNN (3.09M) | 94.203% | 95.362% | 95.217% | 93.333% | 93.478% |

**Table 5.** After adversarial training, Adversarial Attack Performance of MLP alternatives ensemble modules for SEVIT (VIT + Ensemble modules) via Majority Voting. Here FGSM, PGD, BIM, and AutoPGD attack samples are having perturbation budget equal to 0.03

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Ensemble Model** | **Clean Samples** | **FGSM** | **PGD** | **BIM** | **AutoPGD** | **C&W** |
| ViT | - | - | - | - | - | - |
| MLP | 95.652% | 86.377% | 84.493% | 95.652% | 84.058% | 95.652% |
| CNN | 96.087% | 89.420% | 88.696% | 96.087% | 89.710% | 96.087% |
| ResNet-FT | 96.232% | 87.826% | 81.159% | 96.232% | 80.145% | 96.232% |
| ResNet-TL | 93.333% | 84.348% | 80.580% | 93.333% | 78.841% | 93.333% |
| ResNet-FT-CNN | 96.522% | 86.667% | 80.580% | 96.522% | 75.652% | 96.522% |
| ResNet-TL-CNN | 95.217% | 87.391% | 83.333% | 95.217% | 83.333% | 95.217% |

**Table 5\_1.** Adversarial Attack performance of surrogate and SEVIT models after adversarial training (with MLP alternatives) on attack samples via Majority Voting, where these attack samples are generated on the surrogate model through model extraction attack.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Ensemble Model** | **Clean Samples** | **FGSM** | **PGD** | **BIM** | **AutoPGD** | **C&W** |
| CNN Surrogate | **90.71%** | **90.29%** | **90.00%** | **92.03%** | **90.43%** | **92.03%** |
| MLP | 95.652% | 93.768% | 93.913% | 95.652% | 94.348% | 95.652% |
| CNN | 96.087% | 94.928% | 95.072% | 96.087% | 95.362% | 96.087% |
| ResNet-FT | 96.232% | 95.072% | 95.072% | 96.232% | 95.362% | 96.232% |
| ResNet-TL | 93.333% | 91.739% | 92.174% | 93.333% | 91.884% | 93.333% |
| ResNet-FT-CNN | 96.522% | 95.362% | 95.652% | 96.522% | 96.087% | 96.522% |
| ResNet-TL-CNN | 95.217% | 93.913% | 94.203% | 95.217% | 94.493% | 95.217% |

**Distillation**

**Before Adversarial Training-----**

**Table 7.** Distilled Model vs SEVIT-CNN performance (Testing is on Attack Samples generated by SEVIT-CNN)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Clean** | **FGSM** | **PGD** | **AutoPGD** |
| Distilled | 92.57% | 89.13% | 89.42% | 90.14% |
| SEVIT-CNN | 96.232% | 87.391% | 85.217% | 84.928% |

**Table 8.** Model Extraction Attack on Distilled and Original Model (Extracted Model Performance)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Extraction on** | **Clean** | **FGSM** | **PGD** | **AutoPGD** |
| Distilled | 90.29% | 85.07% | 82.46% | 83.33% |
| SEVIT-CNN | 92.14% | 75.65% | 67.54% | 64.93% |

**After Adversarial Training-----**

**Table 9.** Distilled Model vs SEVIT-CNN performance (Testing is on Attack Samples generated by SEVIT-CNN)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Clean** | **FGSM** | **PGD** | **AutoPGD** |
| Distilled | 94.00% | 91.59% | 91.01% | 91.88% |
| SEVIT-CNN | 96.087% | 89.420% | 88.696% | 89.710% |

**Table 10.** Model Extraction Attack on Distilled and Original Model (Extracted Model Performance)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Extraction on** | **Clean** | **FGSM** | **PGD** | **AutoPGD** |
| Distilled | 89.14% | 83.33% | 82.46% | 83.19% |
| SEVIT-CNN | 91.86% | 76.67% | 68.99% | 61.74% |

**SEVIT Pseudocode**

Input: Image I, VIT model with L layers, number of intermediate blocks m

1. Pass the image I through the VIT model to obtain the patch embeddings and positional embeddings:

patch\_embeddings, positional\_embeddings = VIT(I)

2. For each block l = 1 to L, perform the following steps:

a. Use the generated patch embeddings and positional embeddings as input to the l-th block of the VIT model.

b. Obtain the output (class tokens, patch tokens) from the last layer of the i-th block.

c. Pass the patch tokens through an MLP to obtain a prediction for the l-th block.

d. Store the prediction for the l-th block in a list.

3. Obtain the output embeddings (class token) from the last layer of the VIT head.

Pass the class token through an MLP to obtain a prediction for the VIT head.

4. Combine the intermediate block predictions by majority voting to obtain a final prediction:

a. Select the m+1 predictions from the list obtained in Step 2 and Step 3.

b. Compute the mode of the m+1 predictions.

c. Set the final prediction to be the mode.

5. Return the final prediction obtained in Step 4 for the input image I.

**Enhanced SEVIT**

1. Define the SEVIT model architecture with CNN blocks instead of MLP blocks.
2. Train the SEVIT model on the original + adversarial datasets, i.e., adversarial training.
3. Extract the final soft predictions from the SEVIT-CNN model for all images in both datasets.
4. Create a new dataset consisting of the images from both the original and adversarial datasets along with their corresponding soft predictions.
5. Train a new distilled model on the new dataset, where each data point consists of an image and its corresponding soft predictions.
6. Evaluate the performance of the distilled model on the test set and compare it to the performance of the original SEVIT model.