War Project Report

Team Emoji

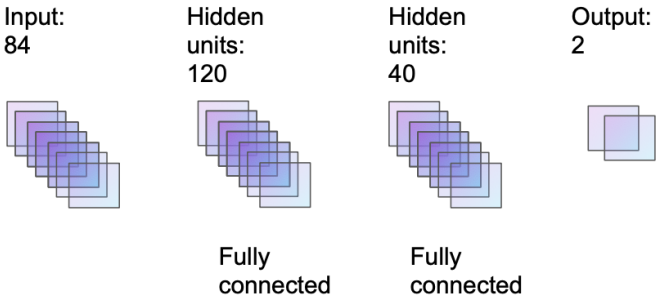
Name: Yifeng Xiong UCI Net ID: yifengx4
Name: Monica Zhou UCI Net ID: quanz13

Our Approach

We focus on the detector in the war project. We trained two models, based on what we have in the defense project. There are two main differences between the defense project and war project. Firstly, instead of the image, we focus on the layers in the LeNet when we input the image. We train a 6-layer LeNet with epoch equals to 35. The first model uses the second last layer of the LeNet as input, and the second model uses the second last layer and the image as input. The second difference is that we use more attacks to perturb the data. We use PGD, and two attacks from other students to perturb the image. We do an ensemble at the end to combine the two detectors together. For levels of robustness, our method can defend against known attacks on unknown images, but it works poorly on unknown attacks.

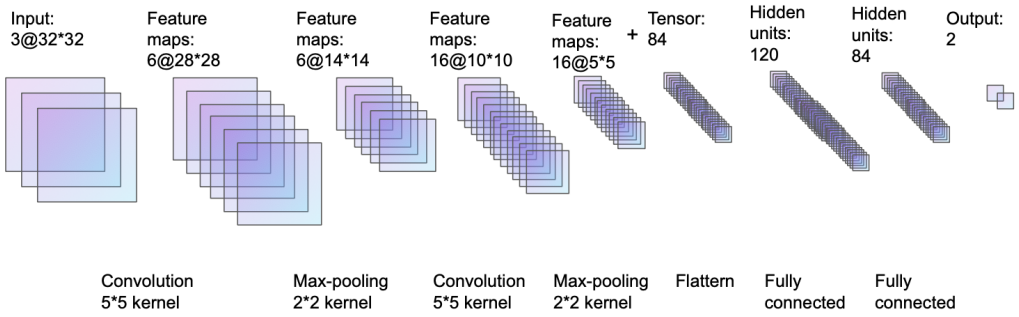
Detector One

The first detector is a neural network with only 3 layers. The input is a tensor, the shape is [1, 84], and the output is an array, and its shape is [1, 2]. Our detector would mark one input as perturbed if the second component is larger and vice versa.



Detector Two

The second detector is a 6-layer LeNet with a little change. We add a tensor with shape [1, 84] to the fourth layer. The output is an array, and its shape is [1, 2]. Our detector would mark one input as perturbed if the second component is larger and vice versa.



Analysis of detector one

Firstly, we would like to determine how much data we should add as a perturbed image. We found that PGD is easy to detect, so we add only 500 PGD perturbed data. For the gibbon and LWY attack, we fix epoch to 8, and then we test several cases:

LWY num	Gibbon num	Clean acc	PGD acc	LWY acc	Gibbon acc
5000	4500	41	85	52	41
5500	4000	28	34	58	57
6000	3500	40	94	58	60
6500	3000	37	81	63	50
7000	2500	11	81	84	84

We pick the model trained by{PGD num = 500, LWY num = 6000, Gibbon num = 3500} as our detector one's model.

Analysis of detector two

For this detector, 500 PGD perturbed data is not enough to provide a convincing result. We first raise the amount to 1000, and then find that 800 PGD perturbed data can provide relatively stable and accurate results. With the amount of PGD perturbed data fixed, we changed the other two perturbed data as the test cases shown below:

PDG num	LWY num	Gibbon num	Clean acc	PGD acc	LWY acc	Gibbon acc
500	3000	6500	34	83	48	61
1000	4000	5000	36	96	58	59
1000	6000	3000	32	96	72	68
800	2000	7200	41	96	59	48

800	6200	3000	41	89	65	64

We pick the model trained by{PGD num = 800, LWY num = 6200, Gibbon num = 3000} as our detector two's model.

Ensemble

Finally, we would like to combine the two detectors together. For each detector, we let them return the last layer, and we get the possibility by the softmax method. With the possibility, we use the following formula to determine whether the image is perturbed or not:
If $\alpha \cdot p_1 + \beta \cdot p_2 \geq \lambda$, then we mark the image as perturbed, and vice versa. To determine α, β, λ, we first notice that our second model is much stronger, and then we try the following cases:

α	β	λ	Clean acc	PGD acc	LWY acc	Gibbon acc
0.5	1.5	1	42	90	59	54
0.4	1.6	1	42	90	57	59
0.3	1.7	1	41	90	63	61
0.2	1.8	1	41	89	66	63
0.1	1.9	1	41	89	64	62

We found that 0.2, 1.8 is the best. Then we try to decrease λ, to decrease a little bit of clean acc, but increase the other three acc, also we change α and β a little bit to get the best result. Finally, we choose: α = 0.23, β = 1.77, λ = 0.95.

Concluding Thoughts

Our model works well locally, and for levels of robustness, our method can defend against known attacks on unknown images, but it cannot work on unknown attacks. With the experience from the defense project, we start the project much quicker, and we schedule a meeting with the Professor to check some methods we may use. With the suggestions from the Professor, we design our new models. We made a powerful attack method, and in the defense project, we made a neural network. It seems our new model still works badly on the unknown attack, this is the limit of our model, with the input perturbed by the given attack, our model can only detect those attacks, and some easy attacks, for example, FGSM. Maybe we should use a more powerful model, like ResNet, but unfortunately, we do not have much time during the final week. In this quarter, we made a very powerful attack method and three detectors. We believe we learned a lot from this course, not only knowledge about adversarial training but also how to read papers and build neural networks. What's more, we get familiar with PyTorch, which we will use a lot in the future. In the end, we would like to thank the Professor and TAs. Thanks for providing us with such an interesting course, although there are some problems due to the first time we cover the material, we still think we enjoy the course.