**초록**

# **적대적 공격에서 살아남기: Data Augmentation & Deep Layers**

|  |  |  |
| --- | --- | --- |
| 이연우 | 정민기 | 한대일 |
| 인공지능응용학과 | 인공지능응용학과 | 인공지능응용학과 |
| 21102376 | 21102385 | 21102398 |

사람의 육안은 쉽게 구별할 수 있지만 딥러닝 모델에게는 구별이 쉽지 않은 데이터를 생성하는 기술을 Adversarial Attack(이하 A.A.)[1]이라고 한다. 이 프로젝트에선 A.A.을 구현하여, 이미지 분류 모델에서 A.A.의 효과를 측정한다. 그리고 A.A.에 대하여, Data Augmentation(이하 D.A.)이나 모델의 깊이 증대 등의 효과적인 방어책을 모색하여 그 성능을 도출한다. 실험 결과, D.A. 중에서 가장 효율적인 방어책은 Random Rotation으로 나타났다. 또한 모델의 층이 증가할수록 A.A.에 대한 방어 능력이 높아진다는 것이 확인되었다.

1. **서론**

A.A.이란 전통적인 가중치 업데이트 방식과 반대되는 방향으로 가중치를 업데이트하여, DNN 이미지 분류 모델의 데이터에 반하는 최악에 가까운 노이즈 이미지를 생성하는 이론이다. Adversarial example은 A.A.을 사용하여 만들어낸 예제로, 육안으로는 쉽게 인식할 수 있지만, 인공지능 모델은 전혀 다른 대상으로 인식하는 이미지를 의미한다.

만일 이러한 기술이 악용된다면 컴퓨터비전 분야의 인공지능 모델에 큰 악영향을 초래할 수 있다. 모델이 이미지를 전혀 다른 라벨로 분류함에도 불구하고 육안으로는 이러한 오류를 일으키는 데이터를 찾아내는 것이 쉽지 않기 때문이다.

따라서 본 프로젝트는 A.A.에 대한 효과적인 방어 방법을 찾아내는 것을 목표로 한다. A.A. 모델로 Fast Gradient Sign Method(FGSM)[2]을 사용할 것이다. FGSM은 신경망 모델이 적대적 변동(adversarial perturbation)을 견디기에는 지나치게 선형적이라는 가정하에, 이미지 분류의 결정 경계를 넘을 수 있는 최소한의 노이즈 분포를 찾는 방법론이다. 이에 대한 방어책으로 몇 가지 D.A. 방법을 시도해 볼 것이다. 또한 모델의 레이어 깊이를 변화시켜 모델의 A.A.에 대한 강건함을 실험할 것이다.

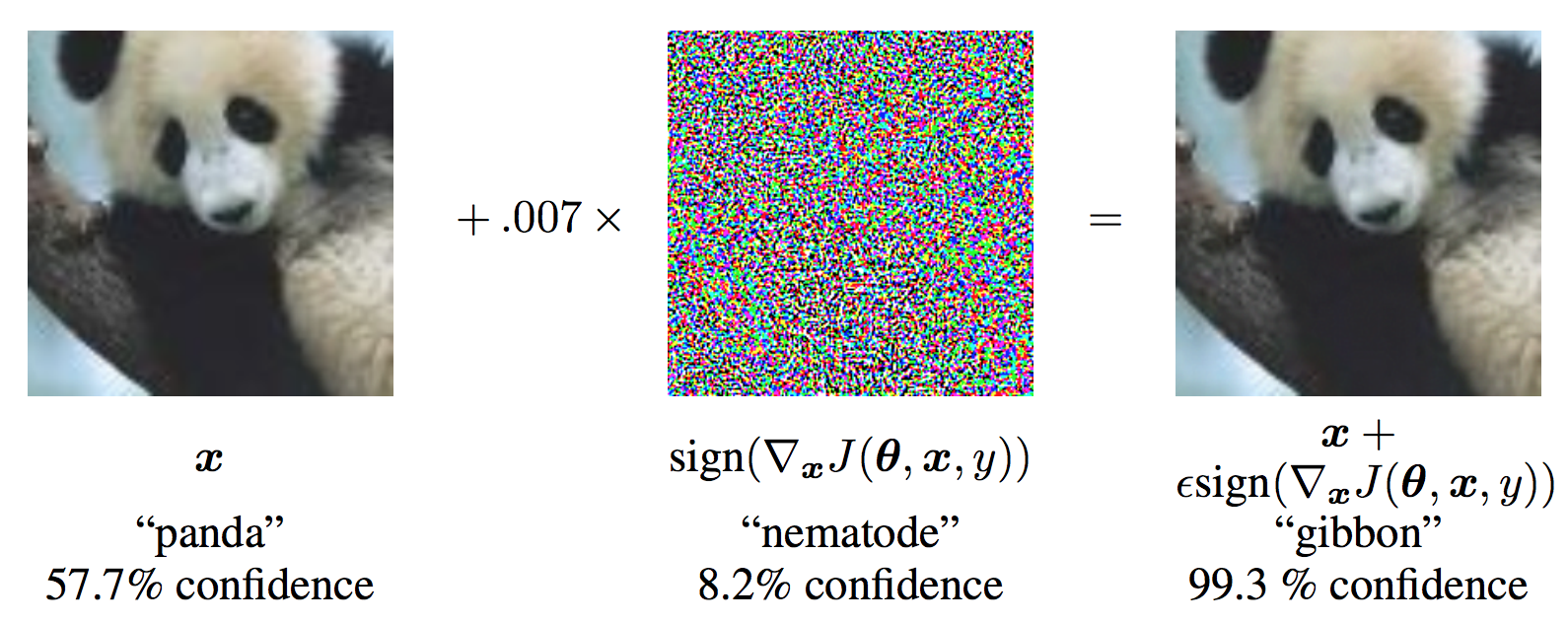
우선 D.A. 방법으로는 Random Horizontal Flip, Random Rotation, Random Affine, Color Jitter, Gaussian Blur를 각각 사용하여 모델을 학습시킨다. 그리고 각각의 모델에 대하여 A.A.을 감행하여, epsilon에 따른 accuracy를 확인할 것이다. 마찬가지로 모델의 깊이를 늘려가며 모델을 학습시킨 뒤, A.A.에 대한 방어 능력을 측정할 것이다.

데이터셋으로는 MNIST 데이터셋과 CIFAR10 데이터셋을 사용한다. 두 데이터셋에 대하여 같은 실험을 반복하여 실험의 일관성을 확인한다.

1. **본론**

FGSM은 가장 널리 쓰이는 A.A. 방법론 중의 하나로, 모델이 그래디언트를 업데이트하는 방향을 고려하여 노이즈를 생성하도록 고안되었다. 즉, 그래디언트를 역전파하는 과정에서 손실을 최대화하는 방향으로 파라미터를 업데이트하도록 고안된 알고리즘이다. 입력 데이터 그래디언트의 손실값을 활용하여 손실을 최대화하는 것이다.

그림 1.



예를 들어 **그림 1.** 을 살펴보면, 57.7%의 신뢰도로 “panda”라고 모델이 분류한 이미지 에 대하여, 우리가 바라는 라벨 , 그리고 모델 파라미터 를 바탕으로 모델을 업데이트하기 위한 손실 을 계산한다. 그리고 입력 데이터의 그래디언트 에 대한 방향을 sign 함수로 나타낸다. 이에 A.A.의 강도라고 볼 수 있는 가중치 을 곱한 뒤, 입력 데이터에 더하면 adversarial example를 생성할 수 있다. 이러한 adversarial example를 모델에 입력할 경우, 모델은 adversarial example를 “panda”가 아닌 “gibbon”으로 분류하게 되는 것이다. 즉, 육안으로는 명백하게 “panda”로 보이는 이미지를 DNN 모델은 전혀 다른 이미지로 분류한다는 것이 A.A.의 의의라고 할 수 있다.

본 프로젝트에서는 FSGM에 대한 방어책으로 D.A.과 네트워크 깊이 증대를 고안하였다. D.A. 방법론으로는 Random Horizontal Flip, Random Rotation, Random Affine, Color Jitter, Gaussian Blur을 선택하였다. Random Horizontal Flip은 주어진 이미지를 임의의 확률에 따라 수평 방향으로 뒤집는 변형 방법이다. Random Rotation이란 주어진 이미지를 임의의 각도로 회전하는 변형 방법이다. Random Affine은 이미지를 랜덤하게 회전 및 이동하는 변형을 의미한다. Color Jitter는 이미지의 명도, 채도, 대비를 랜덤하게 변화시키는 변형을 의미한다. Gaussian Blur는 이미지를 가우시안 분포에 따라 흐릿하게 처리하는 변형 방법을 의미한다.

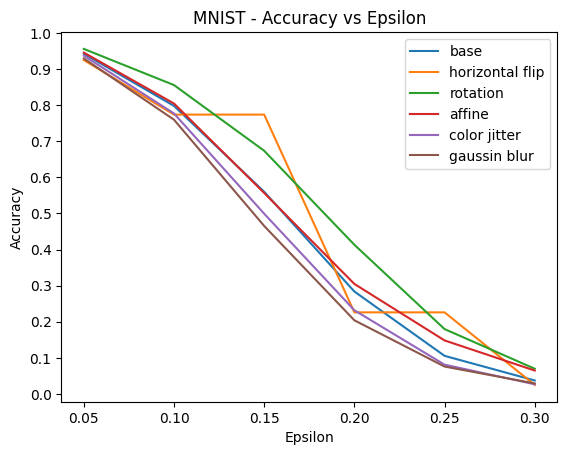
기본적인 학습 모델에 위 D.A. 방법 5가지를 각각 적용한 5개의 모델을 학습한 뒤, 가중치 데이터를 저장한다. 그리고 이렇게 저장된 가중치 데이터에 각각 A.A.을 감행하여 어떠한 D.A. 방법론이 A.A.에 대해 가장 강건한지를 확인한다.

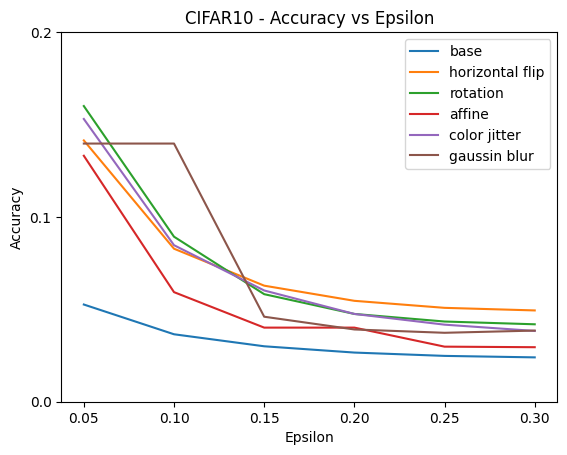
모델은 LeNet-5 모델을 수정하여 사용하였다. 레이어 깊이의 경우, 각각 5층, 7층, 8층 구조로 설계하였다. 레이어의 층 수에 따른 파라미터 수는 각 데이터셋에 대하여 **표 1.** 과 같이 되도록 설계하였다.

표 1 .

|  |  |  |  |
| --- | --- | --- | --- |
| **# Param** | **Net-5** | **Net-7** | **Net-8** |
| **MNIST** | 131,140 | 501,940 | 687,890 |
| **CIFAR10** | 62,006 | 153,662 | 216,406 |

1. **실험 결과**
   1. **D.A. 방법에 따른 모델의 A.A. 방어 능력**





MNIST 데이터셋과 CIFAR10 데이터셋에 대하여 각각 A.A.을 시행하였다.으로 실험하였을 때, FSGM이 각각 잘 동작하는 것을 확인할 수 있다.

표 2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **MNIST** | | **CIFAR10** | |
| **Acc** | **vanilla** | **to A.A.** | **vanilla** | **to A.A.** |
| **BM** | 0.9895 | 0.7976 | 0.7214 | 0.0365 |
| **RHF** | 0.9834 | 0.7736 | 0.7421 | 0.0828 |
| **RR** | 0.9900 | 0.8553 | 0.7251 | 0.0893 |
| **RA** | 0.9907 | 0.8042 | 0.7188 | 0.0593 |
| **CJ** | 0.9904 | 0.7767 | 0.7315 | 0.0847 |
| **GB** | 0.9895 | 0.7584 | 0.7251 | 0.0686 |

*(BM: Base Model, RHF: Random Horizontal Flip, RR: Random Rotation, RA: Random Affine, CJ: Color Jitter, GB: Gaussian Blur)*

**표2.** 는 을 기준으로 실험한 모델들의 정확도를 나타낸 표이다. BM은 D.A.을 적용하지 않은 모델에 대한 정확도를 나타낸다. Vanilla는 A.A.을 시행하지 않은 경우에 대한 정확도를 나타낸다. To A.A.은 A.A.을 시행한 경우 정확도를 나타낸다.

BM의 정확도와 D.A.을 적용한 모델들의 정확도를 비교하였을 때, MNIST 데이터셋의 경우 비교적 큰 차이를 보이지 않는다. 반면, CIFAR10 데이터셋의 경우, A.A.에 대한 방어 능력이라고 할 수 있는 to A.A.에 대하여, BM에서보다 D.A.을 적용한 모델에서 확연이 높음을 확인할 수 있다.

MNIST 데이터셋에서 이미 모델이 데이터셋에 대한 충분한 학습이 이루어졌기 때문에, D.A.가 정확도의 향상을 달성하지 못하고 있다. 그러므로 A.A.가 시행된 경우에도, 모델이 D.A.로부터 유의미한 성능 향상을 얻지 못했기 때문에 A.A.에 대한 가시적인 방어 능력을 획득하지 못한 것으로 보인다.

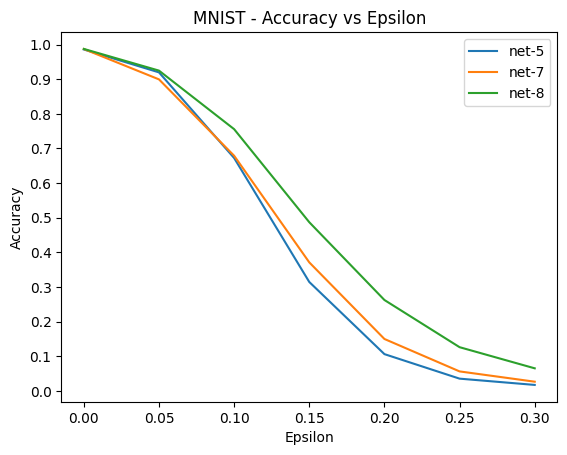
각 D.A. 방법에 대한 방어 능력을 비교해 보면, Random Rotation이 가장 강건함을 확인할 수 있다. BM의 정확도가 그다지 높지 않음에도 불구하고, Acc to A.A.이 가장 높은 수치를 보이기 때문이다.

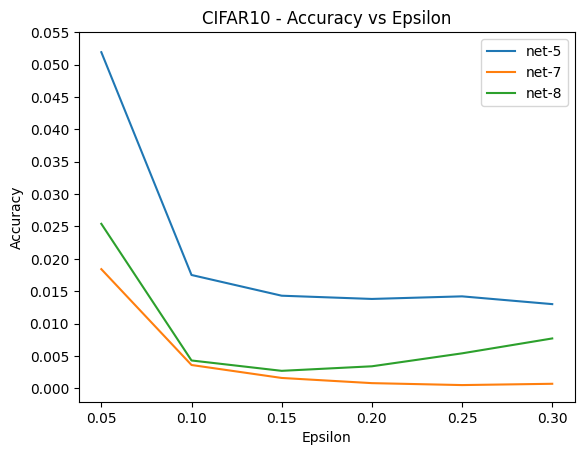
* 1. **모델의 깊이에 따른 A.A. 방어 능력 비교**

표 3.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **MNIST** | | | **CIFAR10** | | |
| **Net-5** | **Net-6** | **Net-7** | **Net-5** | **Net-6** | **Net-7** |
| **0** | 0.9857 | 0.9870 | 0.9870 | 0.5514 | 0.5599 | 0.5546 |
| **0.05** | 0.9196 | 0.8994 | 0.9247 | 0.0519 | 0.0184 | 0.0254 |
| **0.1** | 0.6725 | 0.6790 | 0.7557 | 0.0175 | 0.0036 | 0.0043 |
| **0.15** | 0.3146 | 0.3712 | 0.4871 | 0.0143 | 0.0016 | 0.0027 |
| **0.2** | 0.1064 | 0.1501 | 0.2626 | 0.0138 | 0.0008 | 0.0034 |
| **0.25** | 0.0356 | 0.0565 | 0.1264 | 0.0142 | 0.0000 | 0.0054 |
| **0.3** | 0.0175 | 0.0268 | 0.0654 | 0.0013 | 0.0000 | 0.0077 |

**표 3.** 은 각 데이터셋과 각 모델에 대하여 A.A.의 epsilon에 대항하는 모델의 정확도를 나타낸 표이다. 레이어 깊이에 따른 모델의 방어 능력을 고려할 때, MNIST 데이터셋 모델의 경우 파라미터 수에 따른 정확도 증가가 뚜렷하게 비례함을 확인할 수 있다. 반면 CIFAR10 데이터셋 모델의 경우, 정확도의 증가가 그다지 뚜렷하지 않음을 확인할 수 있는데 이는 모델이 과소적합 되었기 때문이라고 의심된다.





1. **결론**

본 프로젝트에서는 대표적인 A.A. 방법인 FSGM에 대하여 D.A.과 레이어 깊이에 따른 방어 능력을 실험하였다. D.A.의 경우 A.A.에 대해 유의미한 방어 능력이 확인되었으며, Random Rotation 알고리즘이 가장 효율적인 방법인 것으로 측정되었다. 각각의 데이터셋에 대한 적절한 파라미터를 계산하는 것은 추가적인 실험이 필요한 것으로 보이나 파라미터의 개수를 증가시키는 것 역시 효과적인 방어 방법으로 확인된다.

**참고 문헌**

[1] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." arXiv preprint arXiv:1412.6572 (2014).

[2] LeCun, Yann, et al. "Gradient-based learning applied to document recognition." Proceedings of the IEEE 86.11 (1998): 2278-2324.

[3] “Adversarial Example Generation - PyTorch Tutorials 1.13.0+cu117 documentation” <https://pytorch.org/tutorials/beginner/fgsm_tutorial.html?highlight=adversarial+attack>

**GitHub**

https://github.com/moordo91/ml2-project/tree/main/ML2\_Project\_Team6