Triplet Loss를 이용한 Adversarial Attack 연구

오택완*, 문봉교*
*동국대학교 컴퓨터공학과
e-mail : otw8863@dongguk.edu, bkmoon@dongguk.edu

A Study on Adversarial Attack Using Triplet loss

Taek-Wan Oh* , Bong-Kyo Moon*
*Dept of Computer Engineering, Dong-guk University

요 약

최근 많은 영역에 딥러닝이 활용되고 있다. 특히 CNN과 같은 아키텍처는 얼굴인식과 같은 이미지 분류 분야에서 활용된다. 이러한 딥러닝 기술을 완전한 기술로서 활용할 수 있는지에 대한 연구가 이뤄져왔다. 관련 연구로 PGD(Projected Gradient Descent) 공격이 존재한다. 해당 공격을 이용하여 원본 이미지에 노이즈를 더해주게 되면, 수정된 이미지는 전혀 다른 클래스로 분류되게 된다. 본 연구에서 기존의 FGSM(Fast gradient sign method) 공격기법에 Triplet loss를 활용한 Adversarial 공격 모델을 제안 및 구현하였다. 제안된 공격 모델은 간단한 시나리오를 기반으로 검증하였고 해당 결과를 분석하였다.

1. 서론

CNN 아키텍처는 많은 영역에서 활용되고 있다. 특히 이미지 분류 분야에서 많이 사용되고 있다. 그러나 이러한 구조가 악의적 의도를 갖는 공격으로부터 안전한지 물어보면, 안전하다 단정할 수 없다. 오히려 CNN은 작은 noise에도 정상적인 분류를 못하고 오작동한다. 오작동 유발시키는 noise를 원본 이미지에 추가하여 공격하는 기법을 Adverarial Attack이라 부른다, 특히 PGD(Projected Gradient Descent)이라는 공격을 활용하여 MINST 숫자 인식 CNN에 모델에서 숫자 이미지에 노이즈를 추가하여 원래 인식되어야할 숫자가 아닌 다른 숫자로 인식하도록 하는 연구[1]가 이뤄져왔다.

본 연구에서는 기존에 알려진 CNN 모델 공격에 대해 간략히 정리하고, 새롭게 Triplet loss를 활용한 공격모델을 제안 및 구현하여 검증한다.

2. 배경 지식

1) FGSM 공격기법

적의적 공격에서 사용되는 기본 아이디어는 제한된 noise 크기 한도 내에서 잘못된 분류를 하도록 noise를 생성하여 원본 이미지에 합쳐주는 방식이 사용된다.

$$f(x) \neq f(x+\xi), \quad \| \xi \| \leq \epsilon$$

Adversarial Attack 사용되는 기본적인 기법은 FGSM(Fast gradient sign method)공격이다. 해당 공격은 아래와 같은 수식을 통해 공격이 진행된다[1].

$$x^* = x + \epsilon \cdot sign(\nabla_x J(x, y_{true}))$$

여기서 $J(x, y_{true})$는 원본 x를 입력으로 넣었을 때의 y값과 원본이미지 x의 Cross-Entropy loss값이다. 해당 공격은 학습이 수렴되는 방향의 반대 방향으로 노이즈를 생성하여 원본이미지에 더한다. 본래 분류되어야할 클래스가 아닌 다른 클래스로 분류하도록 오작동 시키는 직관적인 방법이다.

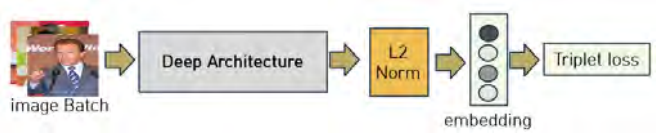
이를 응용하여 확장시킨 방법이 Targeted FGSM이다. 해당 공격은 원하는 타겟으로 이미지의 방향으로 수렴하도록 원본 이미지에 노이즈를 가하는 기법이다. 한 번에 타겟 이미지로 수렴하게 할 수 없기 때문에 여러 스텝 반복하여 공격을 수행하게 되며, 이를 정리한 수식이 아래의 수식이다.

$$x^*_{t+1} = \Pi_{x+S}(x^{*_t} - \epsilon \cdot sign(\nabla_x J(x^{*_t}, y_{target})))$$

위의 수식 공격은 마지막 layer에서는 Softmax등으로 나오는 값에 경우에 효율적인 공격이다. 최근 이미지 분류는 마지막 레이어의 Embedding 값을 구하고 이러한 Embedding간의 거리를 기반으로 클러스터링하여 분류한다. 때문에 기존의 FGSM 공격을 그대로 활용하기 힘들다. 본 연구에서는 공격자 이미지의 Embedding값이 피해자 이미지의 Embedding값에 가까워지도록 적용하기 위해 아래에 기술한 Triplet loss를 활용하여 FGSM을 수행하였다.

2) FaceNet 아키텍처 및 Triplet loss

Triplet Loss를 이용한 Adversarial 공격을 수행하기 위해 FaceNet를 사용했다. FaceNet은 아래의 [그림 1]과 같은 구조를 갖는다. Covolution Layer들을 거친 뒤, L2 Normalize 이후 고차원 이미지를 저차원으로 바꾼 Embedding값을 구한다[2]. 이를 Triplet loss를 기반으로 모델을 학습시키고, 거리를 기반으로 모델을 분류하는 방식을 갖는다.

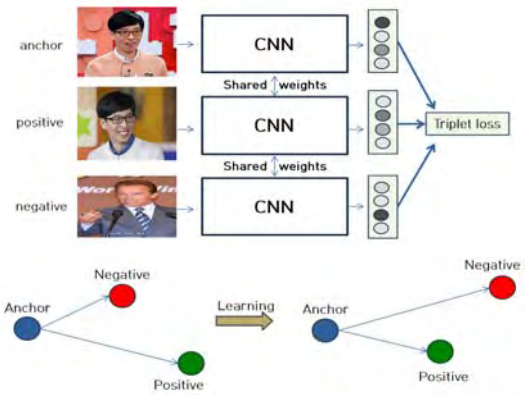


[그림 1] FaceNet 아키텍처

Triplet Loss를 anchor, positive, negative Embedding 값을 받아 임베딩 값 사이의 거리를 조절해주는 손실함수이다.

$$L(f(x_{anchor}),\ f(x_{positive}),\ f(x_{negative}))$$
$$L=\max(0,\alpha+d(f(x_{anchor}),f(x_{pos}))-d(f(x_{anchor}),f(x_{negative})))$$

위의 수식과 같이 표현되며, positive와의 거리가, negative와의 거리보다 α보다 가깝게 하는 방식으로 학습된다.



[그림 2] triplet loss 학습

본 연구에서는 위의 triplet loss와 기존 FGSM의 알고리즘을 기반으로 Targeted Adversarial 공격을 수행했다.

3. 공격모델 제안

공격자의 변경할 얼굴이미지를 x_{adv}, 피해자의 얼굴이미지들을 $x_{victim\ imgs}$, 마지막으로 공격자의 얼굴이미지들을 $x_{attacker\ imgs}$로 정의한다. 공격자의 변경할 얼굴은 기존의 공격자의 embedding 값으로부터 멀어져야하며, 피해자의 얼굴 embedding 값으로부터 가까워져한다. 때문에 공격을 위한 Triplet loss값은 다음과 같이 정의할 수 있다.

$$L(anchor,\ positive,\ negative)$$
$$=L(f(x_{adv}),f(x_{victim\ imgs}),f(x_{attacker\ imgs}))$$

이에 따라 제안하는 Adverarial Attack을 위한 수식은 아래와 같이 정의할 수 있다.

$$x^*_{t+1}=\Pi_{x+S}(x^{*_t}-\epsilon\ \bullet\ sign(\nabla_x L(adv,\ victim,\ attacker)))$$

이때 Triplet loss의 임베딩간 거리 기준인 α는 1.0으로 설정하였다.

$$d(f(x_{anchor}),f(x_{negative})))-d(f(x_{anchor}),f(x_{pos}))>\alpha(=1.0)$$

이외에 공격 성공률을 높이기 위한 5가지 기법을 추가하였다. 첫 번째는 Triplet loss의 Positive와 Negative에 하나 embedding값이 아니라 배치형태로 여러 embedding 값을 사용하였다. 공격자와 피해자의 여러 이미지를 embedding화 시킨 뒤, 각 루프마다 3개를 랜덤하게 추출하여 배치 형태로 값을 전달하였다. 이를 통해 공격자와 피해자 사진 1장만이 아니라 여러 이미지를 바탕으로 노이즈를 생성하기 때문에 보편적인 공격이 가능해 질 것을 기대할 수 있다. 두 번째는 원본이미지와 빠르게 멀어질 수 있도록 무작위의 작은 값을 노이즈로 추가하는 것이다. 세 번째는 이미지의 무작위 영역의 크기를 조금씩 왜곡시켜 원본 이미지로부터 Embedding 값이 빠르게 멀어지게 하였다. 네 번째는 공격의 가속화를 위해 Momentum 기법까지 적용한다[3]. 마지막으로 공격과정에서 변하는 이미지의 값이 이전 이미지와의 차이가 크게 나지 않도록 값을 clip 해준다[4]. 이를 구현한 알고리즘은 아래와 같다.

Algorithm 1 MI-FGSM(using Triplet Loss)

g_0=0; x^*_0=x;

$emb_{attackers}$= get embedding value list of Attacker images

$emb_{victims}$= get embedding value list of victim images

for t=0 to T-1 **do**

$subEmb_{attacker}$= select three, randomly from $emb_{attackers}$

$subEmb_{victim}$= select three, randomly from $emb_{victims}$

Apply image distortion to x_t

x_t+=random_value(minval=-1e-1, maxval=1e-1)

input x^*_t to f and obtain the gradient

$$\nabla_x \text{Loss}(f(x^*_t),subEmb_{victim},\ subEmb_{attacker})$$

$$g_{t+1}=\mu\ \bullet\ g_t+\frac{\nabla_x L(adv,\ subEmb_{victim},\ subEmb_{attacker})}{\|\nabla_x L(adv,\ subEmb_{victim},\ subEmb_{attacker})\|}$$

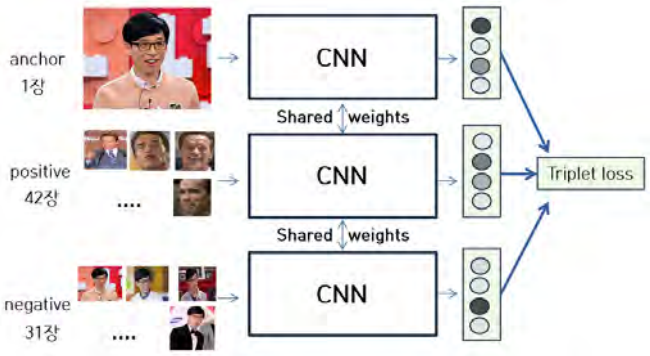
$$x^*_{t+1}=clip(x^*_t+\alpha\ \bullet\ sign(g_{t+1}),\ x^*_t-\epsilon,\ x^*_t+\epsilon)$$

return $x^*=x^*_T$

4. 공격 모델 구현 및 공격 시나리오

제안한 공격 모델 알고리즘을 검증하기 위해, Python, TensorFlow, OpenCV, FaceNet을 사용하여 코드를 구현하였다.

공격의 궁극적인 목표는 공격자의 이미지에 노이즈를 추가하여 피해자의 이미지에 유사하게 인식하도록 하는 것이다. 공격자는 유재석으로 정의하였고, 피해자는 아놀드 슈왈처로 정의하였다. 공격자의 이미지는 총 31장을 사용하였고, 피해자의 이미지는 42장을 사용하였다. 공격을 수행하여 변형시킬 공격자의 이미지 1장의 embedding 값을 Triplet loss의 anchor 값으로, 피해자의 얼굴 이미지 42장의 embedding 값들은 positive 값으로, 공격자의 이미지31장의 embedding 값들은 negative로 두어 공격을 수행했다.



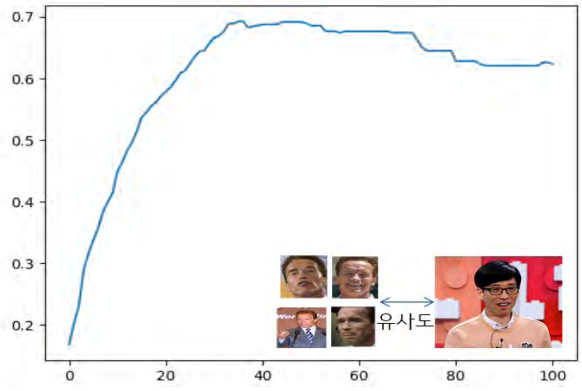
[그림 3] 공격 수행을 위한 Triplet Loss

5. 실험결과

구현한 알고리즘을 수행하면, 아래의 [그림 4]와 같이 기존의 얼굴에 노이즈가 추가된 이미지가 생성된다.

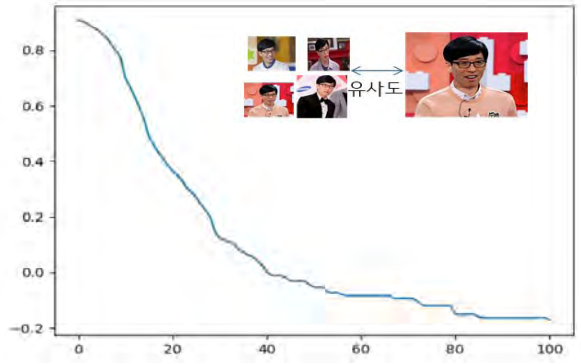
[그림 4] 원본 이미지(좌), 공격수행 이미지(우)

변경된 이미지와 원본 이미지와의 유사도 및 희생자(아놀드)와의 유사도를 비교해본 결과는 아래의 [그림 5]의 그래프와 같다. FaceNet을 거쳐 나온 embedding 유사도가 약 0.16에서 최대 0.691까지 유사도가 오른 것을 확인할 수 있다. 약 70프로의 유사도를 지니기 때문에 악의적 목적을 가지고 공격을 가했을 때, 피해를 줄 수 있는 여지가 존재한다.



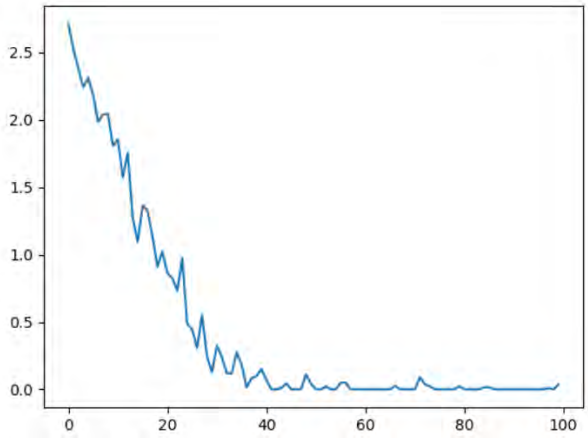
[그림 5] Adversarial 이미지와 아놀드의 유사도

원본 이미지와의 비교를 살펴보면 embedding 값 유사도가 초기 0.91 유사도에서 최저 - 0.14까지 내려간 것을 확인 할 수 있다. 생성된 Adversarial 이미지를 FaceNet에 입력으로 줄 때, 더 이상 유재석으로 분류하지 못함을 보여준다.



[그림 6] Adversarial 이미지와 유재석의 유사도

Adversarial 공격을 수행함에 따라 원본 유재석의 embedding 값과 거리가 멀어지고, 아놀드의 embedding 값 거리와 가까워짐에 따라 Loss 값이 빠르게 감소하는 그래프를 볼 수 있다.



[그림 7] Triplet Loss 그래프

Triplet Loss는 40회 이후부터는 아래의 수식 조건을 만족하기 때문에 loss값이 거의 0으로만 나오게 된다.

$$d(f(x_{anchor}),f(x_{negative}))) - d(f(x_{anchor}),f(x_{pos})) > \alpha (= 1.0)$$

때문에 피해자의 Embedding 값에 일정 거리 이상으로 더 가까워지지 못하게 된다. 이러한 한계점을 개선하기 위해서는 Triplet loss의 Postive embedding과 Negative embedding 값을 랜덤하게 추출하는 것이 아니라 최대한 Triplet loss의 손실이 발생하지 않는 방향으로 샘플링해 주는 것이 필요할 것으로 보인다. 다른 해결 방안으로는 Lossless triplet loss[5]를 사용하는 방법이다. 마지막 embedding 레이어의 Activate Function을 sigmoid로 정의한 뒤, Triplet loss를 아래와 같은 수식으로 적용하는 방식이다.

$$\sum_{i=1}^{N}[-\log(-\frac{d(anchor, positive)}{\beta}+1+\epsilon) \\ -\log(-\frac{N-d(anchor, negative)}{\beta}+1+\epsilon)]$$

이를 활용하면, Loss값이 0이 나오지 않아 특별한 샘플링 과정 없이도 공격 효율을 높일 수 있을 것으로 예상된다.

6. 결론

본 연구는 Triplet Loss를 활용한 Adversarial Attack 공격모델을 제시하고, 이에 대한 코드 구현 및 검증을 연구하였다. 실험 결과에서처럼 Triplet Loss손실 및 특정 유사도 이상으로는 증가하지 않는 한계점이 존재한다. 그러나 공격 수행을 통해 변형된 이미지가 약 70프로까지의 피해자 얼굴 이미지와의 유사도까지 증가하였다는 것은 실제 공격에 활용될 수 있는 여지가 있다고 판단된다. 특히 오픈소스로 CNN 구조를 사용하게 되면, 본 연구에서 제시한 공격 모델을 활용한 공격에 취약할 수 있다.

때문에 오픈소스 CNN구조를 사용하는 경우 더 견고한 모델을 만들기 위해 학습 과정에서 위와 같이 적의적 이미지까지 같이 학습시키는 것이 바람직 할 것으로 생각된다..

참고문헌

[1] Aleksander Madry, “Towards Deep Learning Models Resistant to Adversarial Attacks” - arXiv:1706.06083 - 09 Nov 2017

[2] Florian Schroff, “FaceNet: A Unified Embedding for Face Recognition and Clustering” - IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2015 - 17 Jun 2015

[3] Yinpeng Dong, “Boosting Adversarial Attacks with Momentum” - arXiv:1710.06081 - 22 Mar 2018

[4] Cihang Xie, “Improving Transferability of Adversarial Examples with Input Diversity” - ECCV 2018 - arXiv:1803.06978 - 11 Jun 2018

[5] Marc-Olivier Arsenault, “Lossless triplet loss(A more efficient loss function)” -“https://towardsdatascience.com/lossless-triplet-loss-7e932f990b24”