한국정보전자통신기술학회논문지(jkiiect)'19-12 Vol.12 No.6
논문 19-12-06-629 http://dx.doi.org/10.17661/jkiiect.2019.12.6.629

임의의 잡음 신호 추가를 활용한 적대적으로 생성된 이미지 데이터셋 탐지 방안에 대한 연구

황정환*, 윤지원**

Random Noise Addition for Detecting Adversarially Generated Image Dataset

Jeonghwan Hwang*, Ji Won Yoon**

요 약 여러 분야에서 사용되는 이미지 분류를 위한 딥러닝(Deep Learning) 모델은 오류 역전파 방법을 통해 미분을 구현하고 미분 값을 통해 예측 상의 오류를 학습한다. 엄청난 계산량을 향상된 계산 능력으로 해결하여, 복잡하게 설계된 모델에서도 파라미터의 전역 (혹은 국소) 최적점을 찾을 수 있다는 것이 장점이다. 하지만 정교하게 계산된 데이터를 만들어내면 이 딥러닝 모델을 '속여' 모델의 예측 정확도와 같은 성능을 저하시킬 수 있다. 이렇게 생성된 적대적 사례는 딥러닝을 저해할 수 있을 뿐 아니라, 사람의 눈으로는 쉽게 발견할 수 없도록 정교하게 계산되어 있다. 본 연구에서는 임의의 잡음 신호를 추가하는 방법을 통해 적대적으로 생성된 이미지 데이터셋을 탐지하는 방안을 제안한다. 임의의 잡음 신호를 추가하였을 때 일반적인 데이터셋은 예측 정확도가 거의 변하지 않는 반면, 적대적 데이터셋의 예측 정확도는 크게 변한다는 특성을 이용한다. 실험은 공격 기법(FGSM, Saliency Map)과 잡음 신호의 세기 수준(픽셀 최댓값 255 기준 0-19) 두 가지 변수를 독립 변수로 설정하고 임의의 잡음 신호를 추가하였을 때의 예측 정확도 차이를 종속 변수로 설정하여 시뮬레이션을 진행하였다. 각 변수별로 일반적 데이터셋과 적대적 데이터셋을 구분하는 탐지 역치를 도출하였으며, 이 탐지 역치를 통해 적대적 데이터셋을 탐지할 수 있었다.

Abstract In Deep Learning models derivative is implemented by error back-propagation which enables the model to learn the error and update parameters. It can find the global (or local) optimal points of parameters even in the complex models taking advantage of a huge improvement in computing power. However, deliberately generated data points can 'fool' models and degrade the performance such as prediction accuracy. Not only these adversarial examples reduce the performance but also these examples are not easily detectable with human's eyes. In this work, we propose the method to detect adversarial datasets with random noise addition. We exploit the fact that when random noise is added, prediction accuracy of non-adversarial dataset remains almost unchanged, but that of adversarial dataset changes. We set attack methods (FGSM, Saliency Map) and noise level (0-19 with max pixel value 255) as independent variables and difference of prediction accuracy when noise was added as dependent variable in a simulation experiment. We have succeeded in extracting the threshold that separates non-adversarial and adversarial dataset. We detected the adversarial dataset using this threshold.

Key Words : Adversarial examples, Adversarial attack detection, Convolutional neural network, Deep learning, Random noise addition

*School of Cybersecurity, Korea University
**Corresponding Author : School of Cybersecurity, Korea University (jiwon_yoon@korea.ac.kr)
Received November 20, 2019 Revised November 28, 2019 Accepted December 6, 2019

1. 서론

딥러닝 모델은 이미지 분류 및 객체 탐지 분야에서 활발히 연구되고 있다[1,2,3,4]. 이미지를 다루는 딥러닝 모델은 여러 산업에 적용되어 직접 의사결정을 내리거나 결정을 내리는데 간접적인 정보를 제공하도록 사용되고 있다. 최근에는 자율주행 자동차의 상황 인식의 핵심적인 기술로 사용되고 있다[5].

딥러닝 모델은 기본적으로 오류 역전파(Error back-propagation) 방식을 통해 미분을 하고 오류를 학습한다. 이와 같이 오류의 기울기(gradient)를 학습하는 딥러닝 모델은 [6,7,8,9]에서 보이는 것과 같이 사람의 눈으로 발견하기 어려운, 성능에 오류를 만들어내는 오류를 계산할 수 있다. 이렇게 정교하게 생성된 오류는 딥러닝 모델에 대한 적대적 사례(adversarial examples)를 생성하는데 쓰여 공격(성능 저하)으로 연결될 수 있다. 여러 논문에서 제기된 것과 같이 블랙 박스처럼 내부를 아는 것이 사실상 불가능한 딥러닝의 특성은 이러한 공격을 탐지하는 것을 어렵게 만든다.

모델이 (자율주행 자동차에서와 같이) 의사 결정을 직접 내리거나 (윤리, 생명 등의) 그에 준하는 매우 중요한 업무를 수행할 경우, 공격을 탐지하지 못 하는 딥러닝 모델의 활용은 매우 제한적일 수밖에 없다. [10, 11, 12, 13, 14]에서 보이는 것과 같이 적대적 사례 탐지에 대한 연구가 이루어지고 있다.

본 연구에서는 적대적 사례 공격의 특성을 활용하여 적대적 사례 데이터셋을 탐지하는 방법을 보인다.

2. 선행 연구

2.1 딥러닝 모델에 대한 적대적 사례 공격

적대적 사례 공격은 목표로 하는 딥러닝 모델의 성능 저하(예측 오탐율, 미탐율 증가, 정확도 하락 등)를 위해 입력 데이터에 오류를 더하는 공격이다. 사용자 및 모델 개발자 등 사람의 육안으로는 인식이 되지 않거나 큰 변화가 느껴지지 않지만, 기계는 전혀 다른 데이터로 인식하게 된다.

그림 1. 적대적 사례 예시 : (실제-예측)
(거미-금붕어) (키보드-금붕어) (금붕어-거미)
(거미-키보드) (키보드-거미) (금붕어-키보드)
Fig. 1. Adversarial samples (class-predicted)
(BW-GF) (KB-GF) (GF-BW)
(BW-KB) (KB-BW) (GF-KB)

일반적인 적대적 사례 공격의 정형적 기술은 다음과 같다:

분류기 $f: R^m \rightarrow \{1,2,\cdots,k\}$ 가 m 차원의 이미지를 k 개의 클래스로 분류하는 작업을 한다면, $loss_f: R^m \times \{1,2,\cdots,k\} \rightarrow R^+$ 은 학습 과정에서의 손실 함수(loss function)이라고 할 수 있다. 이 때 공격에 사용되는 적대적 사례는 $f(x) \neq l$ 인 입력데이터 x와 목표 클래스 l에 대해 $x+e \in [0,1]^m$를 만족하는 오류 $\arg\min_e (c\|e\|_2 + loss_f(x+e,l))$을 찾아 입력 데이터 x에 더한 값이다. 사람이 인식하기 어려운 정도를 반영해야 하기 때문에 오류의 크기는 작아야 하고 그럼에도 분류되는 클래스를 바꿀 정도가 되어야하기 때문에 최적화 문제를 푸는 것으로 볼 수 있다.

적대적 사례를 만들어내는 연구에서는 이러한 최적화 문제의 계산량을 줄여 공격을 실행가능하도록 하였다. 본 연구에서는 대표적인 두 가지 공격을 탐지하였다. 첫번째는 오류의 기울기가 가지는 부호 (+/-)만을 반영하여 (사람이 인식하기 어려운) 충분히 작은 값에 곱해주는 것만으로도 효율적으로 공격 사례를 만들어 내는 방법이다. 이 방법을 Fast Gradient Sign Meth-od(FGSM) 공격이라고 부른다[8]. 두번째는 자코비안 행렬을 이용

하여 클래스에 가장 영향을 많이 주는 픽셀을 입력 데이터로부터 찾아내 오류를 주입하는 공격인 Saliency Map Attack이다[9].

2.2 적대적 사례 탐지 연구

2.2.1 데이터의 분포 기반

학습 과정에서의 학습 데이터의 분포와 입력 데이터의 분포가 유사해야 한다는 가정을 포함하고 있다. 데이터를 분포를 계산하는 방식에 따라 탐지 연구를 분류할 수 있다.

커널 밀도 추정 방법은 마지막 활성화 계층의 데이터 벡터를 커널화하여 커널 공간상의 밀도를 추정한다. 이 때 입력 데이터가 학습 데이터의 밀도가 낮은 곳에 위치하는 경우 적대적 사례로 탐지하게 된다[10].

마찬가지로, 데이터 분포를 계산하기 위하여 마할라노비스 거리를 사용한 연구에서는 마할라노비스 거리가 멀어 분포로부터 벗어났다고 판단되는 입력 데이터를 적대적 사례로 탐지한다[11].

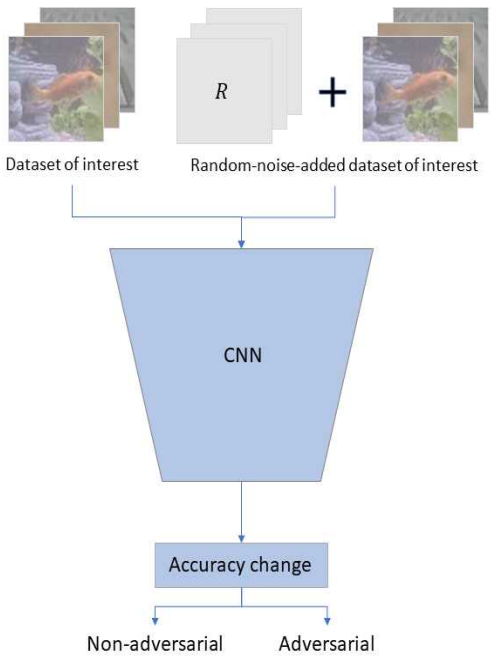

그림 2. 적대적 데이터셋 탐지 프로세스
Fig. 2. Process for detecting adversarial dataset

2.2.2 데이터의 특성 기반

일반적인 문제에 적용하기 보다는 데이터가 가지는 특성을 통해 특정 문제를 해결하는 연구로 적대적 얼굴 데이터를 탐지하는 연구가 있다. 사람의 얼굴에 일반적으로 존재하는 특성(attribute)들을 입력 데이터가 지켜야할 특성이라고 판단하는 Attribute-steered 모델을 제안하였다[12]. 이러한 탐지 기법은 활용되는 수행 업무가 일정 형식을 갖추고 있는 데이터를 활용하거나, 수행 업무에 맞게 특성을 만들어낼 수 있을 경우에 활용할 수 있다.

표 1. 적대적 데이터셋 탐지 알고리즘
Table 1. Detect adversarial dataset algorithm

Algorithm
Detect Adversarial Dataset

- **require** : Classifier f, Model parameter θ,
  Input dataset X,
  Detection threshold T
- **return** : Dataset is adversarial or not

1
$N \leftarrow$ the number of data points in dataset X,
// $X = (X_1, X_2, \cdots, X_N)$
$r \leftarrow$ random number matrix whose shape is the same as each data point $X_i (i \in \{1,2,\cdots,N\})$
$R \leftarrow (r, r, \cdots, r)$
// concatenate r N times

2
$acc_{input} \leftarrow f(X, \theta)$
$acc_{noised} \leftarrow f(X+R, \theta)$

3
if $|acc_{input} - acc_{noised}| > T$:
X is adversarial
else :
X is not adversarial

표 3. 테스트 데이터셋과 공격 기법에 따른 적대적 데이터셋 분류 결과
Table 3. Classification prediction on test dataset and adversarial datasets

Attack	Predicted / Observed	Keyboard	Gold fish	Black widow
- (Original)	Keyboard	44	3	3
	Gold fish	5	45	0
	Black widow	5	0	45
FGSM	Keyboard	2	19	29
	Gold fish	49	0	1
	Black widow	49	1	0
Saliency MAP	Keyboard	0	21	29
	Gold fish	32	0	18
	Black widow	48	2	0

2.2.3 입력 데이터 복원 기반

입력 데이터에 사람이 오류의 영향력을 줄이기 위해 Down Sampling 후 데이터를 복원하거나 변분 오토인코더를 이용해 잠재 변수 공간으로 전환한 후 오류의 영향력을 줄인 채로 원본 입력 데이터와 유사하게 복원한다[13,14].

3. 제안하는 방법

본 연구에서는 주어진 데이터셋이 적대적 데이터셋인지 탐지하는 방법을 제안한다.

해결하고자 하는 문제에 대한 정의는 다음과 같다: 이미지 입력 데이터셋 X가 주어졌을 경우, 이 데이터셋이 적대적인 사례들로 구성되었는지 탐지한다. 이 때 X는 모두 적대적인 데이터로만 이루어져 있거나, 적법한 데이터(오류가 추가되지 않은 관찰된 데이터셋 그대로인 데이터)로만 구성되어 있다고 가정한다.

본 연구에서는 이러한 오류의 영향력을 줄이는 방법으로서 임의의 수를 다시 더해주는 것만으로도 충분하다는 것을 실험적으로 보인다. 다음과 같은 알고리즘을 통해 X가 적대적 데이터셋인지 탐지한다.

표 1의 알고리즘에서 r은 데이터셋의 하나의 데이터 포인트, 하나의 이미지와 같은 모양의 임의의 수 행렬이다. 하나의 이미지의 크기가 $H \times W \times 3(R,G,B)$ 일 경우, r의 모양도

표 2. CNN 구조
Table 2. CNN Architecture

Layer(type)	Output shape	# Params
input layer	(64, 64, 3)	-
conv2d	(62, 62, 32)	896
conv2d	(60, 60, 32)	9248
max pooling 2d	(30, 30, 32)	0
conv2d	(28, 28, 64)	18496
conv2d	(26, 26, 64)	36928
max pooling 2d	(13, 13, 64)	0
flatten	(1, 10816)	0
dense	(1, 1024)	11076608
dense	(1, 1024)	1049600
dense	(1, 100)	102500
output	(1, 3)	303

$H \times W \times 3$이며, 이 때 r의 한 요소는 $|r_{h,w}| \leq noise\,level \leq 255, r_{h,w} \in Z$ 의 범위를 갖는다. 추가되는 잡음의 수준을 나타내는 변수 $noise\,level$ 은 이미지 픽셀의 범위인 255(1)보다 작으며 10(10/255) 이하의 작은 값으로도 충분히 탐지를 수행할 수 있었다. R은 r을 N번 이어 붙인 X와 같은 모양의 임의의 수 행렬이다.

다음과 같은 이유로 위의 알고리즘은 작동한다:

1) X가 적대적 데이터셋이 아닐 경우

입력데이터셋에 더해지는 R은 분류기 f의 성능을 크게 바꾸지 않는다. 적대적 사례를 만들어 내는 과정

을 보면, (분류기의 성능을 저하시키는 기울기의 방향 및 부호만을 계산하기 때문에) 가장 효율적으로 수행되는 공격 중 하나인 FGSM을 보더라도 $(-1,0,1)^{HW}$ 의 공간에서 성공할 수 있는 공격을 찾아야 하므로 임의의 r이 적대적인 사례를 다수 만들어 분류기의 성능을 크게 저하시킬 확률은 $\frac{c}{3^{HWN}}$,(c는 상수) 이하이다. 예를 들어, 픽셀 수 64×64,데이터 개수가 $N = 1000$인 데이터셋에 대하여 임의의 수 R이 더해져 분류기의 성능이 크게 저하될 확률은 $\frac{1}{3^{64 \times 64 \times 1000}}$과 같다.

2) X가 적대적 데이터셋일 경우

모든 적대적 공격은 사람의 눈으로는 발견되지 않아야 한다는 조건을 만족시키도록 구현된다. 따라서, 잡음 수준이 일정 수준 이상일 경우, $X+R$은 정교하게 계산된 오류의 영향력을 없애게 되어 데이터셋의 일부가 적대적 공격이 일어나기 전과 같은 클래스로 분류되게 된다.

4. 실험 및 결과

4.1 실험 조건

4.1.1 실험 환경

본 연구에서 적대적 사례를 만들어 내는 것과 제안한 방법으로 적대적 데이터셋을 탐지하는 것 모두 다음의 동일한 환경에서 진행하였다. 운영체제는 ubuntu 16.04 LTS 이며 CPU는 Intel Core i7-7700, 메모리는 32GB, GPU는 GeForce 1070 을 활용하였다. 적대적 데이터셋을 만들어내는 라이브러리는 foolbox를 활용하였다[15].

4.1.2 데이터셋

Tiny Imagenet 데이터셋은 이미지 분류를 위해 이미지 클래스별 학습, 검증, 테스트 데이터가 존재하며, 본 연구에서는 3가지 클래스('computer keyboard', 'goldfish', 'black widow')를 활용하였다.

4.1.3 이미지 분류 딥러닝 모델

이미지 데이터의 개수 및 클래스 수가 많지 않아, 표 2와 같은 구조의 기본적인 CNN 모델을 사용하였다. 이미지 상의 패턴을 찾아내기 위해 두 번의 컨볼루션 연산 이후 맥스 풀링 연산을 한다. 이 과정을 두 번 반복한 후, 완전 연결 계층을 세 번 추가하여 분류기를 구성하였으며, 출력 계층을 추가하였다. 풀링을 제외한 계층 뒤에 활성화 함수로 relu를 사용하였으며 출력 계층의 활성화 함수는 softmax를 사용하였다.

4.1.4 적대적 사례 생성 공격

공격은 FGSM 공격과 Saliency Map 공격에 대해 적대적 사례 탐지를 수행하였다.

공격은 표 3과 같이 150개의 테스트 데이터에 대해 생성하였으며, 각 공격기법 별로 분류 정확도는 1퍼센트 이하로, 성공적으로 CNN 모델을 속여 성능을 저하시킨 것을 확인할 수 있다.

4.1.5 독립 변수와 종속 변수 설정

실험의 독립 변수는 공격 종류와 잡음 수준이다. FGSM, Saliency Map 두 가지 공격으로 만들어진 적대적 데이터셋에 대해 실험하였으며, 잡음 수준은 RGB 픽셀 값을 (최대 픽셀 값 255 기준) 0부터 19까지 설정하였다. 종속 변수는 임의의 수가 더해진 전후의 예측 정확도의 차이이다. 실험의 신

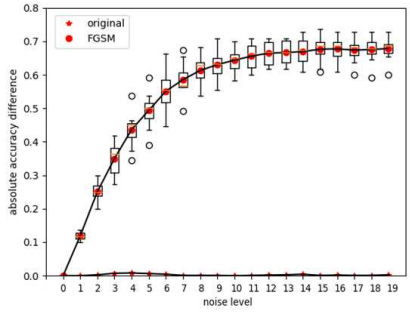

그림 3. FGSM 공격 탐지
Fig. 3. Detection of FGSM attack

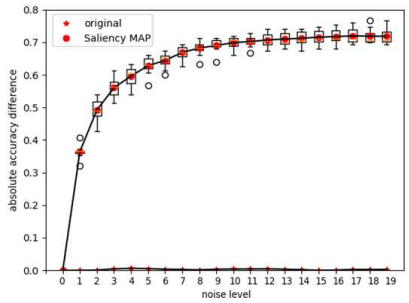

그림 4. Saliency Map 공격 탐지
Fig. 4. Detection of Saliency Map Attack

뢰도를 위하여 임의의 수는 10번의 시뮬레이션 과정을 통해 추출하였으며 결과상의 상자 도표로 확인할 수 있다.

4.2 잡음 수준에 따른 탐지 결과

각 공격기법 및 잡음 수준 별 탐지 결과는 그림 3, 4와 같다. x 축은 잡음 수준이며, y 축은 제안한 탐지 기법 적용 전후의 예측 정확도의 차이의 절댓값(정확도 1 기준)이다.

상자 도표 결과는 일관적이라고 할 수 있다. 잡음 수준이 0에서 19까지 변하는 동안 정상적인 데이터셋의 분류 예측 정확도는 거의 변하지 않은 반면, 적대적인 데이터셋의 분류 예측 정확도는 상당히 많이 변해, 높은 잡음 수준의 경우 평균적으로 70퍼센트 정도까지 변한 것을 알 수 있다.

이를 통해 탐지 역치 T 의 값을 정할 수 있다. T를 정하는 방법은 탐지하고자 하는 공격별, 더해지는 잡음 수준 별로 상이하다. FGSM 방법의 경우 이상치까지 포함하여 모든 시뮬레이션에서 적대적 데이터셋을 탐지할 수 있는 T의 최댓값은 0.1이었으며, 잡음 수준 6 이상의 시뮬레이션에서 적대적 데이터셋을 탐지할 수 있는 T의 최댓값은 0.446이었다. Saliency Map의 이상치까지 포함하여 모든 시뮬레이션에서 적대적 데이터셋을 탐지할 수 있는 T의 최댓값은 0.32이었으며, 잡음 수준 6 이상의 시뮬레이션에서 적대적 데이터셋을 탐지할 수 있는 T의 최댓값은 0.601이었다.

5. 결론

본 연구에서는 딥러닝 모델에 가해질 수 있는 적대적 사례 공격을 탐지하는 방법을 보였다. Tiny Imagenet 데이터셋에서 이미지를 예측 분류하는 딥러닝 모델의 성능을 저하시키기 위해 이미지를 크게 바꾸지 않을 정도의 작은 오류를 미리 계산하여 적대적 사례데이터셋을 생성하였다. 적대적 사례가 만들어지는 과정 및 특성을 반영하여, 이 공격을 탐지하기 위하여 임의의 수를 더하는 방식을 제안하였다. 임의의 수가 더해짐에 따라 예측의 정확도가 변하는 정도가 미리 지정한 탐지 역치를 넘는지의 여부를 통해 적대적 데이터셋을 탐지하였다. 실험 상 최대 잡음 수준에서 적대적 데이터셋의 경우 예측 정확도가 평균 72.1%까지 변하였음을 확인하였다.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” Advances in neural information processing systems, pp. 1097-1105, 2012.

[2] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556, 2014.

[3] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed,D. Anguelov, D. Erhan, V. Vanhoucke,

and A. Rabinovich, "Going deeper with convolutions," Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1-9, 2015.
[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778, 2016.
[5] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, et al., "End to end learning for self-driving cars," arXiv preprint arXiv:1604.07316, 2016.
[6] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," arXiv preprint arXiv:1312.6199, 2013.
[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. WardeFarley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," Advances in neural information processing systems, pp. 2672-2680, 2014.
[8] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," arXiv preprint arXiv:1412.6572, 2014.
[9] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," 2016 IEEE European Symposium on Security and Privacy (EuroS&P), pp. 372-387, IEEE, 2016.
[10] R. Feinman, R. R. Curtin, S. Shintre, and A. B. Gardner, "Detecting adversarial samples from artifacts," arXiv preprint arXiv:1703.00410, 2017.
[11] K. Lee, K. Lee, H. Lee, and J. Shin, "A simple unified framework for detecting out-of-distribution samples and adversarial attacks," Advances in Neural Information Processing Systems, pp. 7167-7177, 2018
[12] G. Tao, S. Ma, Y. Liu, and X. Zhang, "Attacks meet interpretability: Attribute-steered detection of adversarial samples," Advances in Neural Information Processing Systems, pp. 7717-7728, 2018.
[13] B. Liang, H. Li, M. Su, X. Li, W. Shi, and X. Wang, "Detecting adversarial image examples in deep neural networks with adaptive noise reduction," IEEE Transactions on Dependable and Secure Computing, 2018.
[14] U. Hwang, J. Park, H. Jang, S. Yoon, and N. I. Cho, "Puvae: A variational autoencoder to purify adversarial examples," arXiv preprint arXiv:1903.00585, 2019.
[15] J. Rauber, W. Brendel, and M. Bethge, "Foolbox: A python toolbox to benchmark the robustness of machine learning models," arXiv preprint arXiv:1707.04131, 2017.

저자약력

황 정 환(Jeonghwan Hwang) **[정회원]**

- 2018년 고려대학교 융합보안학과 졸업
- 2018년-현재 고려대학교 정보보호대학원 정보보호학과 재학(석사과정)

〈관심분야〉 기계학습, 베이지안 기법, 딥러닝

윤 지 원(Ji Won Yoon) **[정회원]**

- 2008년 University of Cambridge 통계신호처리 박사
- 2011-2012년 IBM Research Lab
- 2012-현재 고려대학교 정보보호대학원 교수
- 2016-현재 정보보호학회 이사
- 2016.6-현재 서울경찰청 사이버보안 자문위원
- 2019.10-현재 국립전파연구원 전파보안 자문위원

〈관심분야〉 통계신호처리, 베이지안 기법, 인공지능