Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.32, NO.5, Oct. 2022 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2022.32.5.997

악성코드 변종 분석을 위한 AI 모델의 Robust 수준 측정 및 개선 연구*

이 은 규,[1†] 정 시 온,[2] 이 현 우,[2] 이 태 진[3‡]
[1,2,3]호서대학교 (대학원생, 학생, 교수)

A Study on Robustness Evaluation and Improvement of AI Model for Malware Variation Analysis*

Eun-gyu Lee,[1†] Si-on Jeong,[2] Hyun-woo Lee,[2] Tea-jin Lee[3‡]
[1,2,3]Hoseo University (Graduate student, Student, Professor)

요 약

오늘날 AI(Artificial Intelligence) 기술은 악성코드 분야를 비롯하여 다양한 분야에서 광범위하게 연구되고 있다. 중요한 의사결정 및 자원을 보호하는 역할에 AI 시스템을 도입하기 위해서는 신뢰할 수 있는 AI 모델이어야 한다. 학습 데이터셋에 의존적인 AI 모델은 새로운 공격에 대해서도 견고한지 확인이 필요하다. 공격자는 악성코드를 새로 생성하기보단, 기존에 탐지되었던 악성코드의 변종을 대량 생산하여 공격에 성공하는 악성코드를 탐색한다. AI 모델의 Misclassification을 유도하는 Adversarial attack과 같이 대부분의 공격은 기존 공격에 약간에 변형을 가해 만든 공격들이다. 이러한 변종에도 대응 가능한 Robust한 모델이 필요하며, AI 평가지표로 많이 사용되는 Accuracy, Recall 등으로는 모델의 Robustness 수준을 측정할 수 없다. 본 논문에서는 Adversarial attack 중 하나인 C&W attack을 기반으로 Adversarial sample을 생성하여 Robustness 수준을 측정하고 Adversarial training 을 통해 Robustness 수준을 개선하는 방법을 실험한다. 본 연구의 악성코드 데이터셋 기반 실험을 통해 악성코드 분야에서 해당 제안 방법의 한계 및 가능성을 확인하였다.

ABSTRACT

Today, AI(Artificial Intelligence) technology is being extensively researched in various fields, including the field of malware detection. To introduce AI systems into roles that protect important decisions and resources, it must be a reliable AI model. AI model that dependent on training dataset should be verified to be robust against new attacks. Rather than generating new malware detection, attackers find malware detection that succeed in attacking by mass-producing strains of previously detected malware detection. Most of the attacks, such as adversarial attacks, that lead to misclassification of AI models, are made by slightly modifying past attacks. Robust models that can be defended against these variants is needed, and the Robustness level of the model cannot be evaluated with accuracy and recall, which are widely used as AI evaluation indicators. In this paper, we experiment a framework to evaluate robustness level by generating an adversarial sample based on one of the adversarial attacks, C&W attack, and to improve robustness level through adversarial training. Through experiments based on malware dataset in this study, the limitations and possibilities of the proposed method in the field of malware detection were confirmed.

Keywords: artificial intelligence, robustness, adversarial attack

Received(08. 22. 2022), Modified(09. 22. 2022), Accepted(09. 22. 2022)

* 이 논문은 2022년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임(No.2019-0-00026, 지능화된 악성코드 위협으로부터 ICT 인프라 보호)

† 주저자, legleg1216@gmail.com
‡ 교신저자, kinjecs0@gmail.com(Corresponding author)

I. 서 론

AI(Artificial Intelligence) 기술은 엄청난 양의 데이터에 대한 접근성 및 증가하는 처리 능력의 필요성으로 인해 등장한 주요 기술이다. 이는 악성코드 분야를 비롯하여 이미지 인식, 물체 감지, 자연어 처리, 자동 주행 등 다양한 분야에서 광범위하게 연구되고 있다. AI 모델은 원하는 기능을 제공하기 위해 대규모 데이터셋을 기반으로 학습을 진행한다. 따라서 AI 기반 시스템은 학습 데이터셋의 품질과 대표성에 크게 의존하지만, 그 양은 제한적이다. 훨씬 더 방대한 데이터셋을 쉽게 얻을 수 있다 하더라도, AI 시스템을 운영하면서 발생할, 실 환경에서의 모든 입력을 더 큰 데이터셋으로 대응할 수 있다는 보장을 하기에는 충분치 않다. 더불어, AI를 학습한 데이터는 모두 과거의 데이터이다. AI 모델을 위협하는 다양한 공격 기술의 발전으로 인해 기존 공격들을 학습한 AI 모델이 새로운 공격에도 효과적으로 대응 가능하다는 신뢰성이 부족하다. EU에서 제시한 AI 개발에 대한 지침을 도출하는 7가지 주요 요구사항 중 다수의 항목이 AI의 신뢰성 및 견고성의 요소를 강조하고 있다[1]. 신뢰성 및 견고성을 보장하지 못할 경우, 데이터 보호, 사이버보안, 개인정보 보호 등 중요 자원을 보호하고, 의사결정을 내리는 것에 있어 AI 기술을 사용하는 것에 대한 우려가 존재한다. 신뢰할 수 있는 AI 시스템을 위해서는 앞으로 등장할 새로운 공격들에도 대응 가능한지 견고성을 확인할 수 있고, 설명할 수 있는 AI 모델이 구축되어야 한다.

공격자는 악성코드가 탐지되면, 기존 악성코드의 변종을 대량 생산하며 공격에 성공하는 악성코드를 찾아 공격을 이어나간다. 현재 발생하는 공격 대부분은 이전에 발생하였던 공격에 약간의 변형을 주어 탐지체계를 회피하는 방식으로 발생한다[2]. 공격자들은 새로운 공격 방법을 창작하기보단, 기존의 공격을 약간 변형하여 탐지체계를 회피하는 식의 공격 방법이 효과적이기 때문이다. 악성코드 변종으로부터 대응 가능한 Robust한 AI 모델인지 Robustness 수준을 확인할 수 있어야 한다. 이러한 공격기법 중 AI 모델의 Misclassification, 신뢰도 하락 등을 유도하는 Adversarial attack이 있다. Adversarial attack이란 원본에 쉽게 구별하기 어려운 아주 작은 변조(Perturbation)를 주어서 AI 모델을 속이는 공격 기법이다[3]. AI 모델의 탐지를 회피하기 위해 Adversarial attack에 대한 관심이 증가하는 추세이다. Adversarial attack이 2014년 Szegedy[4] 등에 의해 소개된 이후 Fig. 1과 같이 Adversarial attack에 관한 사례 및 공격 관련 논문이 지속적으로 증가하고 있다. 하지만 국가 취약점 데이터베이스를 의미하는 NVD(National Vulnerability Database) 내에 Adversarial attack 관련 항목이 존재하지 않으며, Adversarial attack에 대한 실제 대응 활동은 많지 않다. Adversarial attack을 위한 다양한 공격 기법들이 발전하고 있으며, 취약성을 완화하기 위한 다양한 접근법이 개발되고 있다. 하지만 상당수의 방법이 새로운 공격기법의 발전으로 무너졌다[5]. 이에 따라 Adversarial attack에도 대응 가능한 견고한 AI 모델을 구축할 수 있어야 한다. 견고한 AI 모델은 단순히 Accuracy, Precision, Recall, F1 score 등과 같은 평가지표로 측정하는 것이 아니라, 모델의 견고성인 Robustness 수준의 측정을 통해 신뢰성 확인이 필요하다. 하지만 아직까지 견고성을 평가하기 위한 공식적인 표준화된 방안이 합의되지 않았으며[6], 생성된 AI 모델에 대한 견고성을 확인할 수 있는 검증 방안이 필요하다.

대부분의 Adversarial attack 연구에서 이미지 데이터를 가지고 실험을 진행한다. 본 논문에서는 이미지 데이터가 아닌 악성코드 데이터를 통해 실험을 진행하여 보안 분야에서의 확장 가능성을 확인한다. 악성코드 데이터셋을 이용하여 DNN을 구축하고 Adversarial attack 중 하나인 C&W attack을 기반으로 Adversarial sample 생성 및 Adversarial training을 진행한다. 이후 Robustness 수준 변화를 관찰하고 측정방법 및 제

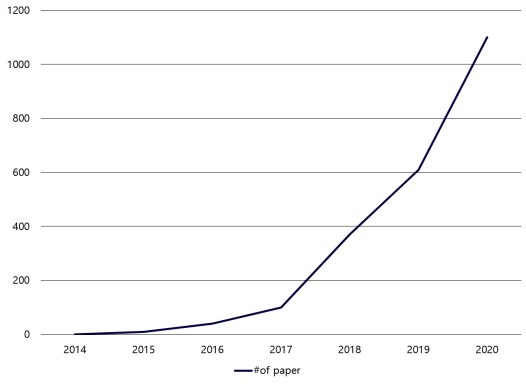

Fig. 1. Papers on adversarial attack in arXiv

안 Framework의 한계점 및 가능성을 확인한다.

본 논문의 구성은 다음과 같다. 2장에서는 AI 모델의 Misclassification을 유발하는 Adversarial Attack의 종류와 해당 공격의 대응방안, 모델의 견고성 평가방안을 비롯한 모델 평가방안들을 소개한다. 3장에서는 본 논문에서 제안하는 모델의 Framework 설명과 더불어 모델의 견고성 평가 산출 방법을 설명한다. 4장에서는 실험에 활용한 데이터셋에 대한 설명과 3장에서 제시한 모델의 견고성 평가방안을 통해 결과를 산출하고, 산출된 성능을 비교한다. 5장에서는 실험 결과에 대한 해석 및 원인에 대해 논의를 하며, 6장에서 결론을 맺는다.

II. 관련 연구

2.1 Adversarial Attack

Adversarial Example은 Fig. 2과 같이 모델이 Training을 통하여 형성된 Decision boundary와 실제 Task boundary의 오차로 인하여 모델이 Misclassification를 일으킬 수 있는 예제를 말한다. Adversarial attack은 Fig. 3과 같이 원본에 쉽게 구별하기 힘든 아주 작은 변형(Perturbation)을 주어 원본이 Adversarial example이 생성될 수 있는 영역으로 가도록 하여 모델의 Misclassification를 유발하며, 신뢰도 하락을 유발하는 공격기법이다. Adversarial attack의 목적은 원본에 최소한의 Perturbation을 가해도 라벨이 변경되어 산출되는 Adversarial sample을 생성함으로써 Perturbation을 쉽게 식별할 수는 없으면서도, 높은 확률로 Misclassification을 발생시키는 공격을 수행하여 서비스 운용에 문제를 유발하는 것이다. 이를 위해서 Perturbation에 Threshold를 지정하여 매우 적은 Perturbation만으로도 Misclassification을 발생시키는 Adversarial sample을 생성할 수 있다. 현재 발생하는 대부분의 공격이 기존 공격을 변형한 공격으로 Adversarial attack이 성격과 일치한다. Adversarial attack을 구현하는 기술은 다음과 같이 다양하다. FGSM(Fast Gradient Sign Method)[4]은 Input gradient와 반대되는 방향으로 Gradient를 조정하여 Adversarial sample을 생성하는 공격기법이다. DeepFool[7]은 더 효율적이고 더 작은 변형(Perturbation)으로 비선형 신경망 구조에 여러 번의 질의를 통해 공격한다. Gradient 계산이 아닌 여러 개의 점에서 Decision boundary에 대해 수직으로 투영하고 적당한 Noise를 추가하는 공격기법이다. JSMA (Jacobian-based Saliency Map Attack)[8]는 Input이 Output에 미치는 변화를 행렬로 Mapping 하여 모델이 Misclassification 하도록 Input gradient를 변화시켜 Adversarial sample을 생성하는 공격기법이다. ZOO Attack (Zeroth Order Optimization based Black-Box Attack)[9]은 기울기 기반 방식과 유사하게 최적화 문제를 해결한다. 하지만 모델의 정보를 알 수 없는 블랙박스 상황을 가정해 기울기를 직접 가져와 사용하는 것이 아닌 기울기를 추정하여 사용하는 최적화 방식이다.

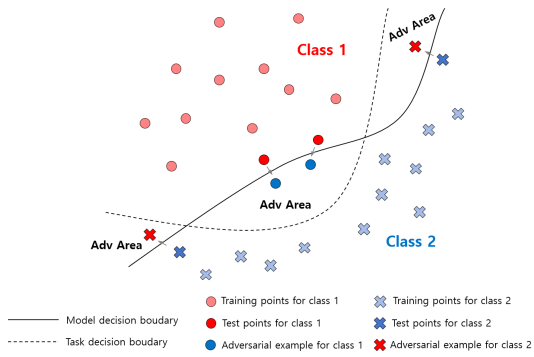

Fig. 2. Adversarial Example

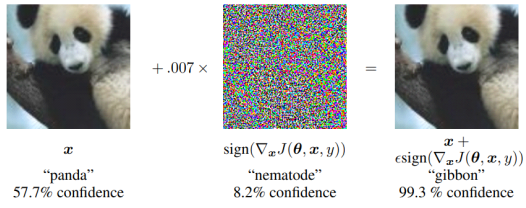

Fig. 3. Adversarial attack[4]

C&W(Carlini&Wagner)[10]는 Approximate 방법으로 최소한의 Noise를 담당하는 손실 함수와 공격 성공률을 높이는 손실 함수의 합을 최소화함으로써 최적의 Adversarial sample을 찾는 공격기법이다. 원본 데이터와 공격의 성공으로 인해 생성된 Adversarial sample 사이의 거리를 측정하는 방법으로는 Linf, L0, L2가 있다. Linf의 경우 변경된 feature가 가장 많이 변경된 값을 말하며, L0의

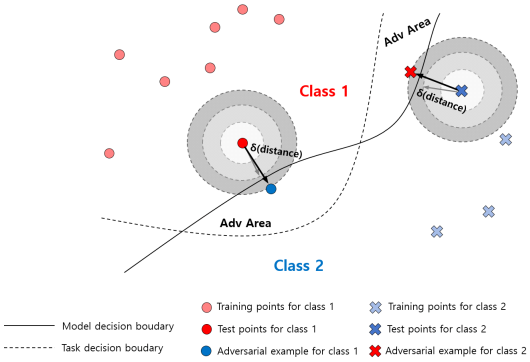

Fig. 4. Adversarial Sample Generation

경우 변경된 feature의 수를 말한다. L2의 경우 원본 데이터와 Adversarial sample을 각각 점으로 표현하였을 때 점과 점 사이의 거리를 유클리디안 알고리즘을 통하여 계산한 값을 말한다. C&W Attack은 다른 Adversarial Attack 기법들에 비하여 L0, L2에서 성능이 뛰어난 것으로 알려져 있으며, 두 가지의 distance metric 중 L2 distance metric이 가장 많이 사용된다. L0 distance의 경우 단순히 원본 데이터와 생성된 Adversarial sample 사이의 변경된 feature 수를 산출하여 사용한다. L2 distance의 경우 수식 (1)과 같이 동작한다.

$$L2 = \left(\sum_{i=1}^{n} | x_i - x_i' |^2 \right)^{\frac{1}{2}} \quad (1)$$

여기서 x는 원본 데이터를 말하며, x'는 생성된 Adversarial sample을 말하며, n은 데이터가 가진 feature의 개수로 정의된다. 즉, L2 distance는 두 점 사이의 오차 제곱 합을 구하는 식이다. Adversarial sample 생성은 Fig. 4와 같이 앞서 산출한 Distance를 파라미터로 사용하여 원본에서 가장 가까우면서도 Misclassification를 유발하는 Adversarial sample을 생성한다.

2.2 Adversarial Attack 대응 방안

Adversarial attack에 대응하기 위해서는 Miscalssification과 신뢰도 하락에 대한 근본적인 해결이 필요하다. Adversarial attack에 대응할 수 있는 기법으로 기울기인 Gradient를 숨기는 방

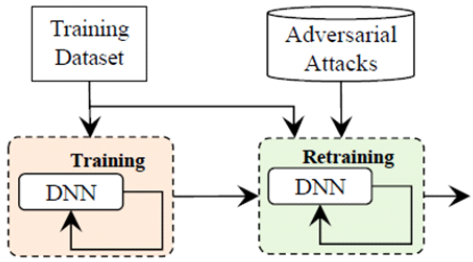

Fig. 5. Adversarial training[3]

식인 Gradient masking, 공격이 될 만한 Noise를 포함하는 Input을 넣지 못하도록 데이터를 정제하는 Input pre-processing이 있다. 그리고 Fig. 5와 같이 공격이 될 만한 데이터를 Adversarial attack을 기반으로 sample을 생성하여 모델 학습에 추가 후 재학습하는 Adversarial training이 있다. Adversarial sample을 통해 비슷한 공격에 대한 대응이 가능하도록 모델을 재학습 시켜 모델의 Robustness 수준의 향상을 기대할 수 있다[3]. Gradient masking의 경우 Gradient를 사용하여 공격하는 기법이 아닐 경우 공격에 대한 대응이 불가능하며, Input pre-processing의 경우 최적화 문제로 인해 최선의 기법인지 확신하기가 어렵다. 따라서 Adversarial training을 통해 모델을 재학습시킬 경우, Adversarial sample 생성을 어렵게 만들어 Robustness 수준 상승을 기대할 수 있다.

Adversarial training을 통해 모델을 재학습시키기 위해서는 Adversarial attack method를 사용하여 Adversarial sample의 생성이 필요하다. 학습 데이터를 대상으로 Perturbation 기반의 Noise를 삽입하여 Misclassificiation 및 신뢰도 하락을 유발하는 Adversarial attack을 진행하여 Adversarial sample을 생성한다. 앞서 소개한 Adversarial attack 방법들은 AI 모델을 속이기 위한 최소한의 변형(Perturbation)을 찾는 기술들이다.

2.3 모델 평가 방법

2.3.1 AI 모델 평가

보편적으로 AI 모델의 성능 평가 중 중요하고 기본이 되는 지표로는 Accuracy, Precision, Recall, F1 score가 있다. Accuracy란 Positive

와 Negative를 정확히 맞춘 비율인 정확도를 의미한다. 이는 전체값들 중 실제 값이 Positive일 때 예측값이 Positive일 경우와 실제 값이 Negative일 때 예측값이 Negative일 경우를 모두 고려한 비율이다. Precision이란 예측값이 Positive인 데이터에서 예측값과 실제값이 Positive인 데이터의 비율로, 정밀도를 의미한다. 실제 값이 Negative이지만 예측값이 Positive인 FP를 낮추는 데 초점을 맞춘다. Recall이란 실제값이 Positive인 데이터들에서 예측값과 실제 값이 Positive로 일치한 데이터의 비율인 재현율을 의미한다. 실제 값이 Positive이지만 예측값이 Negative인 FN을 낮추는 데에 초점을 맞춘다. F1 score란 Precision과 Recall을 결합한 지표이다. Precision과 Recall의 조화평균으로, Precision과 Recall이 어느 한쪽으로 치우치지 않는 수치를 나타낼 때 높은 수치를 갖게 된다.

2.3.2 AI 모델 견고성 평가

위와 같이 Accuracy, Recall 등 AI 모델의 성능을 평가할 수 있는 지표들이 존재하지만, 보편적인 AI 모델 평가 방법들만으로는 모델의 견고성을 증명할 수 없다. 따라서 AI 모델의 견고성을 증명할 수 있는 검증 방안이 필요하다.

Chang[11] 등은 AI 모델의 견고성을 분석하기 위한 채점 방안을 제안하였다. Adversarial Robustness Toolbox[12], Foolbox[13], CleverHans[14] 등의 API를 사용하여 6종류의 CNN 모델과 13종류의 공격기법을 통해 공격 정확도를 산출하여 각 모델에 대한 각 공격 별 score를 산출한 후 각 모델에서의 공격 별 robustness score의 평균에 대하여 표준편차를 산출하였다. 위 방법은 분산이 높을수록 의미 있는 robustness score를 가진다고 판단하는 방식이다.

Berghoff[15] 등은 견고성을 평가하기 위한 검증 방안을 제안하였다. 제안한 방법은 기존의 Perturbation뿐 아니라 Image noise(gaussian noise 등), Pixel perturbations(L0, Linf 등), Geometric transformations(rotation, scaling 등), Colour transformations(hue 등) 총 네 가지 관점에서의 다양한 방식으로 실험을 진행하였으며, 각각의 방법으로 원본 데이터에 작은 값부터 변형을 가하여 어느 정도까지의 큰 변형을 가하더라도 모두 정확하게 모델이 분류하는 sample의 개수로 robustness score를 측정하였다. AI 모델이 정확하게 분류한 sample의 개수를 비율로 환산하여 해당 비율이 높을수록 모델이 견고하다고 판단하는 방식이다.

Hartl[16] 등은 AI 모델에 대한 견고성 검증 방안으로 ARS(Adversarial Robustness Score)를 제안하였다. ARS란, Adversarial attack에 얼마나 Robust한 지를 나타내는 수치이다. ARS 수치 산출 방법은 먼저, 기존 AI 학습 데이터셋에서 데이터 그룹별로 Adversarial sample을 생성한다. 생성된 Adversarial sample과 원본 데이터 간 거리의 대략적인 평균값을 산출하여 해당 그룹의 ARS로 사용한다. Adversarial sample을 생성하기 어려울수록 Noise는 점점 커지고 그에 따라 원본과의 거리인 ARS도 증가하게 된다. ARS가 클수록 Adversarial sample 생성에 어려움이 있는 것이고 이를 통해 Robust한 정도를 확인할 수 있다. Fig. 6는 Hartl[16]등이 산출한 각 공격 타입별 IDS에서의 Recall Score 및 ARS이다. Fig. 6를 보면 IDS가 매우 높은 확률로 공격을 탐지할 수 있는 것을 확인할 수 있다. 하지만 모델이 견고하다는 의미는 현재 존재하는 공격뿐만이 아닌 앞으로 새롭게 등장할 공격에 대해서도 대응할 수 있음을 의미한다. 따라서 기존 Recall, Accuracy와 같은 평가지표만으로는 앞으로 새롭게 등장할 공격에 대한 방어 정도인 모델의 Robustness를 판단하기에는 한계가 있다. ARS는 원본 데이터에 Perturbation을 부여하며 Misclassification를 발생시킬 Adversarial sample의 원본과의 Distance 즉, Perturbation의 정도를 의미한다. 즉, ARS가 낮다는 것은 원본과 새롭게 생성된 Adversarial sample 간의 차이가 적다는 것으로 약간의 Perturbation만 가해줘

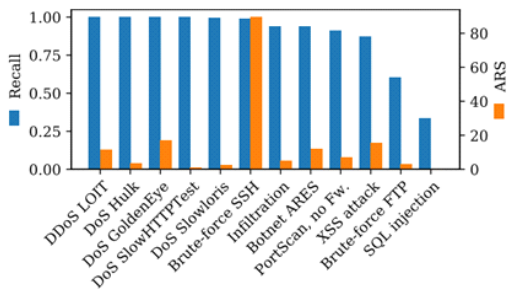

Fig. 6. Recall for unmodified flows and ARS for attacks in CIC-IDS-2017.[16]

도 모델이 Misclassification를 발생시킬 공격을 찾기 쉽다는 것을 의미한다. 즉, 특정 공격에 대한 ARS가 낮다는 것은 모델은 해당 공격에 대한 Robustness를 지니지 못했음을 의미한다. Fig. 6에서 확인 할 수 있듯 'DoS SlowHTTPTest', 'DoS Slowloris', 'DoS Hulk'와 같은 공격들의 Recall Score를 보면 지금 당장은 IDS가 해당 공격들을 잘 탐지할 수 있겠지만, 해당 공격들을 조금만 수정하여 공격하면 IDS는 해당 공격들을 탐지하기 어려울 것이다. 이를 검증하는 실험을 수행하였으며, 실험 결과는 Fig. 7과 같다. Fig. 7은 기존에 잘 알려진 Adversarial Attack 기법인, CW, PGD, FGSM 등을 적용한 후 공격을 수행하였을 때 산출한 Recall Score를 그래프로 나타낸 것이다. 앞서 언급된 Recall Socre는 높지만, ARS는 낮은 공격그룹들을 살펴보면, 아무것도 적용하지 않고 공격을 진행한 'Unmodified'를 보면 Recall Score가 매우 높게 산출되어 해당 공격이 잘 검출되는 것을 확인할 수 있으나 그에 반하여 CW, PGD, FGSM과 같은 Adversarial Attack을 수행하며 Recall Score가 매우 떨어지는 것을 확인할 수 있다. 이렇듯 기존에 잘 알려진 Recall, Accuracy 등과 같은 모델 평가지표로 모델을 평가하는 것은 한계가 존재하기에 앞으로 등장할 공격에 대한 견고성을 산출한 ARS를 모델 평가지표로써 활용하는 것이 적합하다고 판단된다. 따라서 ARS 수준 측정을 통해 모델의 견고한 정도를 파악하고 특정 부분의 Robustness를 보강하기 위한 작업을 통해 견고한 모델의 생성이 가능하다.

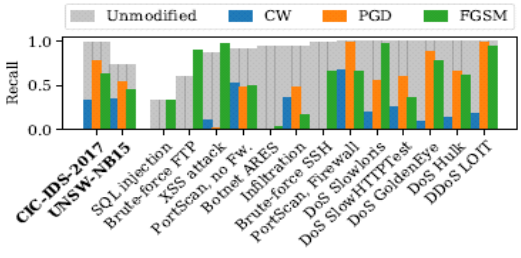

Fig. 7. Attack success ratios for both datasets and per attack type for CIC-IDS-2017.[16]

III. 제안 모델

지속해서 발전하는 악성코드의 특성상 과거의 데이터를 통한 학습은 Adversarial attack과 같이 새로운 변종 공격들에 대해서도 효과적으로 대응 가능한지 확신할 수 없다. 공격자들은 기존 탐지체계를 회피하기 위해 이전에 발생했던 공격들을 기반으로 약간의 변형을 부여한 변종을 생성하는 방식으로 공격을 시도한다. 하지만, AI 모델의 보편적인 평가지표로 사용되는 Accuracy, Recall 등으로 변종 공격들에 대해 대응이 가능한지 평가하는 것은 충분하지 않다. 악성코드 변종들에 대해서도 대응 가능한지 Robustness 수준 측정 및 이를 개선하는 방법이 필요하다.

이 장에서는 제안하는 AI 모델의 Robustness 수준 측정방법 및 수준을 개선하는 방법에 대하여 제안한다. 본 논문에서는 기존 공격을 변형하여 AI 모델을 회피하는 Adversarial attack을 기반으로 측정하는 ARS를 AI 모델의 Robustness 수준 평가지표로 활용한다. 해당 지표를 활용하여 학습 데이터 내의 Robustness 수준이 부족한 그룹을 파악하고 Adversarial training을 통해 Robustness 수준을 강화하고 부족한 학습이 개선되는지 확인하고자 한다.

3.1 Proposed Framework

본 연구에서 제안하는 모델은 Adversarial sample을 생성하는 방법으로 앞서 소개한 Adversarial attack 기법 중 C&W Attack을 사용하였다. C&W Attack의 경우 기존 JSMA와 DeepFool에 비하여 L0 기반 Attack과 L2 기반 Attack에서 좋은 성능을 보인다. 원본과 Adversarial sample에서 변경된 특징 수를 기준으로 하는 L0 기반 Attack 보다, 일반적으로 수식 (1)과 같이 원본과 Sample의 거리를 기준으로 하는 L2 기반 Attack을 많이 사용한다. 본 연구에서는 L2 기반 C&W를 사용하였고, Adversarial sample을 생성하기 위한 수식 (2)와 같다.

$$\begin{aligned} &minimize\ D(x, x+\delta) + c \cdot f(x+\delta) \\ &such\ that\ x+\delta \in [0,1]^n \end{aligned} \tag{2}$$

여기서 $D(\cdot)$는 원본과 생성시킬 Adversarial sample 사이의 거리를 산출하기 위한 Distance metric을 말한다. $f(\cdot)$는 기존의 어려운 접근법을 간단한 수식으로 변환하여 이를 해결하기 위해 제시

한 Objective 함수이다. 이에 대한 자세한 수식은 수식 (3)과 같다.

$$f_6(x') = (\max_{i \neq t}(Z(x')_i) - Z(x')_t)^+ \quad (3)$$

해당 함수를 통하여 Adversarial sample이 정상적으로 의도한 라벨이 산출되었는지를 판단하도록 할 수 있다. 여기서 $Z(x)_i$는 x가 라벨 i로 산출될 확률을 말한다. 만약 Adversarial sample이 의도한 대로 라벨 t(target)로 산출된다면 해당 Objective 함수의 산출 값은 음수가 된다. 이렇듯 C&W는 간단한 수식을 정의함으로써 작업 효율성을 높였다. 즉, C&W Attack을 통하여 산출 라벨이 변경되면서 원본과 가장 가까운 Adversarial sample을 생성할 수 있다.

모델을 생성한 후 모델의 견고성을 검증하는 방안으로는 ARS를 사용하였다. 제안하는 모델은 Fig. 8과 같으며, 해당 모델의 동작 방식은 다음과 같다. 우선 기존에 수집한 학습 데이터셋을 기반으로 AI 모델을 구축한다. 이후 학습 데이터셋을 대상으로 C&W Attack을 이용하여 데이터셋 내의 그룹별 Adversarial sample을 생성한다. 생성한 Adversarial sample을 기존 학습 데이터셋에 추가하여 AI 모델을 재학습하는 Adversarial training을 진행한다. Adversarial training 이후 이전과의 Robustness 수준 비교를 위해 그룹별 데이터를 대상으로 다시 한번 C&W Attack을 통해 Adversarial sample을 생성한다. 이전의 ARS 수치와 Adversarial training 이후의 ARS 수치를 비교하여 ARS 수치가 향상된 것을 확인한다. 개선된 Robustness 수준을 통해 새로운 공격들로부터 향상된 신뢰도를 제공하는 AI 모델을 구축할 수 있음을 볼 수 있다.

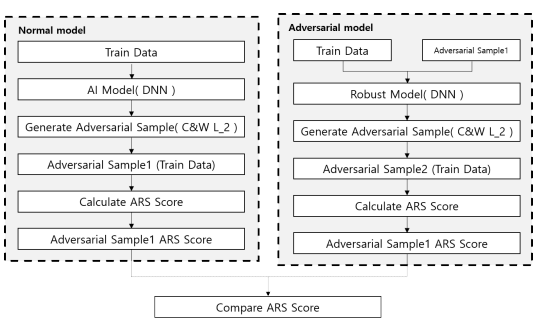

Fig. 8. Proposed Framework

3.2 악성코드 Featuring

Adversarial attack 기법들은 대부분 입력 데이터가 이미지인 AI 모델을 고려하여 개발되었다. 하지만 이미지와 같이 각 feature의 범위가 일정하지 않은 데이터에 동일하게 적용하기에는 부적합하다. 이에 따라 최대-최소 정규화(min-max normalization) 기법을 사용하여 데이터의 feature 값들을 0과 1 사이로 변환하였다.

3.3 ARS 측정 방법

본 연구에서 사용된 ARS 측정 방식은 다음과 같다. L2 기반의 유클리디안 거리 측정 방식을 사용하여 원본 데이터와 Adversarial sample 간의 거리를 측정한다. 측정한 값을 바탕으로 각 그룹별 평균 수치를 산출하며 이때, Adversarial sample 생성에 실패한 데이터는 평균 산출 시 해당 데이터를 제외한 후 각 그룹별 Adversarial sample 생성에 성공한 N개에 대한 평균을 산출한다. ARS 측정 수식은 수식 1과 같다.

$$ARS = \frac{1}{N}\sum_{i=1}^{N} LSUBs_i \in S\, d_{s_i} \quad (4)$$

여기서 S는 각 그룹별 Adversarial sample 생성에 성공한 데이터의 집합을 뜻하며, s는 각 그룹에 속해 있는 Adversarial sample을 뜻한다. 또한 d_s는 원본 데이터와 s에 대한 거릿값을 뜻한다. 즉, ARS를 통해 각 그룹별로 Adversarial sample 생성에 성공한 데이터들을 대상으로 원본 데이터와 Adversarial sample 사이 거릿값들의 평균을 산출한다.

IV. 실험 결과

4.1 Dataset

본 논문에서는 2019 KISA Datachallenge 악성코드 데이터셋을 사용하였다. 데이터셋의 구성

Table 1. Configuration of Dataset

Dataset	Malware	Normal	Total
Train	17,562	11,568	29,130
Test	4,513	4,518	9,031

Table 2. Configuration of AVClass

AVClass	Train	Test
autoit	40	171
ramnit	40	57
scar	40	17
winactivator	40	56
zegost	40	37
Total	200	338

은 Table 1과 같다. 학습 데이터셋은 악성 17,562개, 정상 11,568개 총 29,130개의 데이터를 사용하였으며, 테스트 데이터셋은 악성 4,513개, 정상 4,518개 총 9,031개의 데이터를 사용하였다. 해당 데이터셋에서 확인된 AVClass 약 800가지 중 가장 많이 검출된 상위 5개의 AVClass만을 사용하여 실험을 진행하였다. AVClass의 구성은 Table 2와 같다. 학습 데이터는 autoit 40개, ramnit 40개, scar 40개, winactivator 40개, zegost 40개로 총 200개를 사용하였으며, 테스트 데이터는 autoit 171개, ramnit 57개, scar 17개, winactivator 56개, zegost 37개로 총 388개를 사용하였다.

데이터셋의 Feature 구성은 악성코드의 PE(Portable Executable) 구조를 분석하여 Feature를 추출하였다. PE 파일의 PE header와 PE section에는 파일의 실행에 필요한 정보가 존재한다. 이 중 PE header의 정보에서 37개의 feature를 추출하였고, PE section의 Entropy를 사용하여 128개의 feature로 변환하여, PE 파일로부터 총 165개의 feature를 추출하여 사용하였다.

4.2 실험 결과

4.2.1 AI 성능 분석 결과

AI 모델은 Fig. 9과 같이 여러 Layer로 이루어진 DNN 모델을 사용하였으며, 사용된 Parameter는 과적합을 방지하기 위하여 초반 3개의 Layer에서 Dropout 0.25를 적용하였고 batch_size = 100, epochs = 93을 학습 파라미터로 지정하였다. 활성화 함수는 ReLU을 사용하였으며, 마지막 계층에 2개의 출력으로 Softmax를 사용하여 각 라벨에 대한 확률값이 산출되도록 모델을 설계하였다. AI 모델의 학습 결과는 Table 3과 같다.

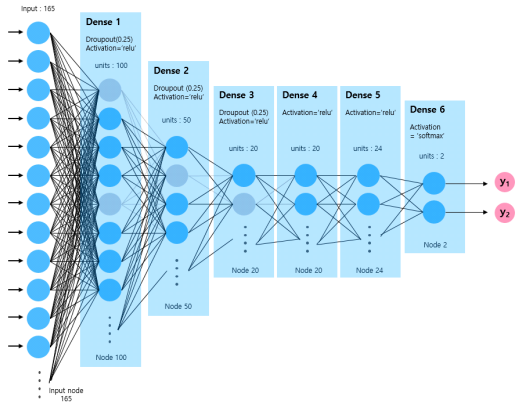

Fig. 9. Proposed DNN Model

Table 3. Results of ai model learning

	Accuracy	Precision	Recall	F1 score
DNN	97.41%	97.42%	97.41%	97.41%

4.2.2 AI Robustness 성능 분석 결과

Adversarial Attack을 위해서 C&W Attack 기법을 사용하였으며, 사용된 C&W Attack의 Parameter는 Table 4와 같다. Adversarial training 이전에 생성한 Adversarial sample의 경우 5개 family 각각에 대해 40개씩 총 200개의 Adversarial sample을 생성하였다. 그 결과 autoit 16개, ramnit 23개, scar 15개, winactivator 39개, zegost 31개의 데이터에 대해 Adversarial sample 생성에 성공하였다. ARS 수준 측정 결과 autoit 0.3650, ramnit 0.7513, scar 4.4400, winactivator 0.6370, zegost 1.0574의 수치를 나타냄을 알 수 있었다. Adversarial training 이후에 생성한 Adversarial sample의 경우 Adversarial training 이전과 같이 5개 family 각각에 대해 40개씩 총 200개의 Adversarial sample을 생성하였다. 그 결과 autoit 16개, ramnit 22개, scar 32개, winactivator 39개, zegost 31개의 데이터

Table 4. C&W attack Parameter

Parameter	Value
Confidence	0.0
Targeted	False
Learning_rate	1.00E-02
binary_search_step	100
max_iter	1000
initial_const	0.01
max_having	500
max_doubling	500
batch_size	1

에 대해 Adversarial sample 생성에 성공하였다. ARS 수준 측정 결과 autoit 0.5765, ramnit 2.3748, scar 0.9097, winactivator 0.6809, zegost 0.4536의 수치를 나타냄을 알 수 있었다.

Adversarial training 이전 모델과 Adversarial training 이후 모델의 5개의 family에 대한 Accuracy를 산출하였다. Accuracy 산출 결과는 Table 5와 같다. Adversarial training 이전 모델에서 autoit 1.0, ramnit 0.98245, scar 0.88235, zegost 1.0, winativator 0.98214가 산출되었다. Adversarial training 이전 모델에서 5개의 AVClass에 대한 평균 Accuracy는 0.99817의 수치를 나타냄을 알 수 있었다. Adversarial training 이후 생성된 모델에서는 autoit 0.99415, ramnit 1.0, scar 0.88235, zegost 0.97297, winativator 0.98214가 산출되었다. Adversarial training 이후 모델에서 5개의 AVClass에 대한 평균 Accuracy는 0.98520의 수치를 나타냄을 알 수 있었다. 산출 결과를 통해 Adversarial training 이후 5개의 AVClass 중 ramnit의 경우 Accuracy가 증가하는 것을 확인할 수 있었으며, scar, winativator에서는 Accuracy가 유지, autoit, zegost은 오히려 Accuracy가 감소하는 경향이 있음을 확인할 수 있었다.

Adversarial training 이전의 ARS 수치와 이후의 ARS 수치를 비교하였다. 그 결과는 Table 6, Fig. 10과 같다. Fig. 10에서 ARS_1은 Adversarial training 이전의 ARS 수치를 의미하며, ARS_2는 Adversarial training 이후의 ARS 수치를 의미한다. family 중 autoit, ramnit, winactivator의 경우에는 ARS 수치가 정상적으로 상승한 것을 확인할 수 있었지만, scar와 zegost의 경우 ARS 수치가 오히려 줄어든 것을 확인할 수 있었다.

Table 5. Comparison of Accuracy Score before and after Adversarial training

	Number of data	Normal model	Robust model
autoit	171	1	0.99415
winativator	56	0.98214	0.98214
scar	17	0.88235	0.88235
zegost	37	1	0.97297
ramnit	57	0.98245	1
total	338	0.98817	0.98520

Table 6. Comparison of ARS before and after Adversarial training

AVClass	autoit	ramnit	scar	winativator	zegost
ARS Before Adverial training	0.3650	0.7513	4.4400	0.3670	1.0574
ARS After Adverial training	0.5765	2.3748	0.9097	0.6809	0.4536

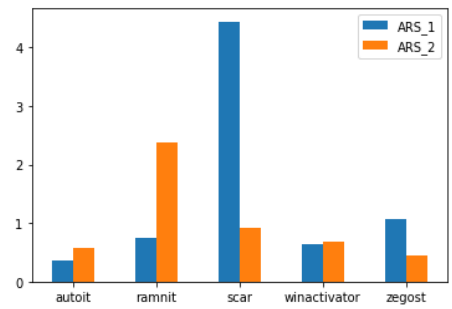

Fig. 10. Comparison of ARS before and after Adversarial training

V. Discussion

Adversarial attack은 한 데이터에 대해 실제 Label과 다른 Label로 AI가 예측하도록 데이터 주변을 탐색하며 적절한 Perturbation을 찾는 공격이다. 하지만 모든 경우의 수를 탐색할 수 없기에 이 문제를 정면으로 해결하는 것은 불가능하다. 따라서 Adversarial attack 방법들은 이 문제를 최적화 문제로 공식화하여 찾아 나간다. 그렇게 찾은 Sample을 재학습하여 해당 원본 데이터로부터

Adversarial attack 성공을 어렵게 함으로써 Robustness 수준을 향상시키는 것에 목적이 있다. 하지만 Adversarial attack으로 찾은 Sample이 원본 데이터와 가장 가깝고 유일한 Sample임을 확신할 수 없다. Adversarial attack을 재수행했을 때, 이전에는 탐색하지 못했던 영역에서 원본과 더 가까운 Sample을 발견할 가능성이 있으며, 다른 영역에도 Sample이 다량 존재할 수 있다. 따라서 고차원의 공간과 탐색에 성공할 확률을 고려하여 원본 데이터에 대한 다수의 Sample을 학습한다면 Robustness 수준 향상이 가능할 것이다. 하지만 한 번의 Adversarial attack에도 많은 연산이 소요되는 특성상 데이터에 반복하여 수행할 수 있는 횟수를 무한정 늘릴 수는 없다. 따라서 향후 한정된 자원과 적절한 Adversarial attack 수준을 고려하는 Framework 연구가 필요해 보인다.

VI. 결 론

AI의 기능 및 역할이 확장됨에 따라 신뢰할 수 있는 AI의 필요성이 대두되고 있다. 중요한 의사결정을 내리기 위해서는 AI에 대한 신뢰성 및 견고성이 확인되어야 한다. 공격자는 공격을 이어나가기 위해 새로운 악성코드를 생성하기보단, 기존 악성코드의 탐지를 회피하기 위해 약간의 변형을 추가하는 형태로 공격을 이어나간다. 이러한 변종에 대해 AI 모델의 견고성 측정 및 부실한 견고성을 개선할 필요가 있다. 따라서 본 논문에서는 신뢰할 수 있는 AI를 위해 AI 모델의 Robustness 수준 측정 및 개선 방법에 대해 제안하였다. 또한, 기존 연구에서 많이 사용되는 이미지 데이터셋이 아닌 실제 악성코드 데이터셋을 기반으로 실험을 진행하여 한계 및 가능성을 확인하였다.

본 논문의 실험에서는 악성코드 데이터셋을 이용하여 DNN을 구축하였다. 이어서 Robustness 수준 측정 및 개선을 위해 C&W attack을 기반으로 Sample 생성 및 Adversarial training을 진행하였다. 실험 결과로는 AVClass에 따른 실험 대상 family인 ramnit, scar, zegost 에서 각기 다른 결과가 산출되었다. ramnit family에서는 처음 의도한 대로 ARS가 증가하는 경향을 보였으나 반대로 scar, zegost에서는 ARS가 감소하는 결과가 산출되었다. 이러한 결과가 산출되는 원인으로는 Adversarial sample 탐색방법의 한계로 인해 발생하는 것으로 파악된다. Adversarial sample 생성은 원본에서 최대한 가까우면서 Label이 다르게 산출되는 지점을 찾는 문제로 직결된다. 이를 해결하기 위해 Adversarial attack을 최적화 문제로 변환하여 원본에서 최대한 가까우면서 라벨이 다르게 산출되는 Adversarial sample을 탐색한다. 이러한 탐색방법은 무한대에 가까운 경우의 수를 효율적으로 탐색할 수 있게 하지만 결국 이미지 데이터가 아닌 Feature가 다양한 악성코드 데이터와 같은 고차원 공간에서 모든 좌표를 완벽하게 탐색할 수는 없다. 처음 탐색을 진행한 방향 외의 영역에서도 여러 Sample들이 존재할 수 있으며, 그중 원본과 더 가까운 거리에 존재하는 Sample은 없다고 확신할 수 없다. 따라서 한 데이터에 대해 견고해지고 싶다면, 그 주위를 여러 번 탐색, 즉 Adversarial attack을 여러 번 수행하여 Sample들을 확보하고 재학습을 진행하여야 한다. 하지만 Adversarial attack을 수행하는 데 많은 연산이 소요되는 특성상 한정된 자원과 적절한 Adversarial attack 수준을 고려하는 Framework에 관한 연구가 필요해 보인다.

References

[1] Hamon, Ronan, Henrik Junklewitz, and Ignacio Sanchez. "Robustness and explainability of artificial intelligence." Publications Office of the European Union, Feb. 2020.

[2] Diro, A. A., & Chilamkurti, N "Distributed attack detection scheme using deep learning approach for Internet of Things," Future Generation Computer Systems 82, pp. 761-768, Feb. 2018.

[3] Shafique, Muhammad, et al. "Robust machine learning systems: Challenges, current trends, perspectives, and the road ahead," IEEE Design & Test 37.2, pp. 30-57, Apr. 2020

[4] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples," arXiv preprint arXiv:1412.6572, Dec.

2014
[5] Biggio, Battista, and Fabio Roli. "Wild patterns: Ten years after the rise of adversarial machine learning," Pattern Recognition 84, pp. 317-331, Jul. 2018
[6] Carlini, Nicholas, et al. "On evaluating adversarial robustness," arXiv preprint arXiv:1902.06705, Feb. 2019.
[7] Moosavi-Dezfooli, Seyed-Mohsen, Alhussein Fawzi, and Pascal Frossard. "Deepfool: a simple and accurate method to fool deep neural networks." Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2574-2582, Jun. 2016.
[8] Papernot, Nicolas, et al. "The limitations of deep learning in adversarial settings," 2016 IEEE European symposium on security and privacy (EuroS&P). IEEE, pp. 372-387, May. 2016.
[9] Chen, Pin-Yu, et al. "Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models," Proceedings of the 10th ACM workshop on artificial intelligence and security, pp. 15-26, Nov. 2017.
[10] Carlini, Nicholas, and David Wagner. "Towards evaluating the robustness of neural networks," 2017 ieee symposium on security and privacy (sp). Ieee, pp. 39-57, Jun. 2017.
[11] Chang, Chih-Ling, et al. "Evaluating robustness of ai models against adversarial attacks," Proceedings of the 1st ACM Workshop on Security and Privacy on Artificial Intelligence. pp. 47-54, Oct. 2020.
[12] Nicolae, Maria-Irina, et al. "Adversarial Robustness Toolbox v1. 0.0," arXiv preprint arXiv:1807.01069. Jul. 2018.
[13] dRauber, Jonas, Wieland Brendel, and Matthias Bethge. "Foolbox: A python toolbox to benchmark the robustness of machine learning models," arXiv preprint arXiv:1707.04131. Jul. 2017.
[14] Papernot, Nicolas, et al. "Technical report on the cleverhans v2. 1.0 adversarial examples library," arXiv preprint arXiv:1610.00768. Oct. 2016.
[15] Berghoff, Christian, et al. "Robustness testing of ai systems: a case study for traffic sign recognition," IFIP International Conference on Artificial Intelligence Applications and Innovations. Springer, Cham, pp. 256-267, Jun. 2021.
[16] Hartl, Alexander, et al. "Explainability and adversarial robustness for rnns," 2020 IEEE Sixth International Conference on Big Data Computing Service and Applications (BigDataService). IEEE, pp. 148-156, Aug. 2020.

〈저자소개〉

이 은 규 (Eun-gyu Lee) 학생회원
2022년 2월: 호서대학교 정보보호학과 졸업
2022년 3월~현재: 호서대학교 정보보호학과 석사과정
〈관심분야〉 악성코드 분석, 침입 탐지, 이상징후 탐지, 정보보호, AI

정 시 온 (Si-on Jeong) 학생회원
2018년 3월~현재: 호서대학교 컴퓨터공학부 학석사과정
〈관심분야〉 악성코드 분석, 정보보호, AI

이 현 우 (Hyun-woo Lee) 학생회원
2017년 3월~현재: 호서대학교 컴퓨터공학부 학석사과정
〈관심분야〉 악성코드 분석, 정보보호, AI

이 태 진 (Tae-jin Lee) 종신회원
2003년 2월: 포항공과대학교 컴퓨터공학과 졸업
2008년 2월: 연세대학교 컴퓨터공학과 석사 졸업
2013년 1월~2017년 2월: 한국 인터넷진흥원 팀장
2017년 2월: 아주대학교 컴퓨터공학과 박사 졸업
2017년 3월~현재: 호서대학교 정보보호학과 교수
〈관심분야〉 시스템 보안, 악성코드 분석, 침해사고 대응, AI