반도체디스플레이기술학회지 제22권 제3호(2023년 9월)
Journal of the Semiconductor & Display Technology, Vol. 22, No. 3. September 2023.

CNN 모델을 이용한 사기 스마트 컨트랙트 탐지

박다은[*]·박용범[**†]
[*]단국대학교 컴퓨터학과, [**†]단국대학교 소프트웨어학과

Fraudulent Smart Contract Detection Using CNN Models

Daeun Park[*] and Young B. Park[†]**
[*]Dankook University Dpt. of Computer Science, [**†]Dankook University Dpt. of Software

ABSTRACT

As the DeFi market continues to expand, fraudulent activities using smart contracts have also increased. HoneyPot and Ponzi schemes are well-known frauds that exploit smart contracts. While several studies have demonstrated the potential to detect smart contracts implementing these scams, there has been a lack of research focusing on simultaneously detecting both types of fraud. This paper addresses this gap by harnessing artificial intelligence to conduct experiments for the detection of both HoneyPot and Ponzi schemes. The study employs the CNN (Convolutional Neural Network) model, commonly used for malware detection. To effectively utilize CNN, the bytecode of smart contracts is transformed into visual representations. The experimental results showcase a recall rate of 0.89 and an F1 score of 0.85, indicating promising detection capabilities.

Key Words : Smart Contract, Fraud, Honeypot, Ponzi, CNN

1. 서 론

최근 인공지능 기술이 발전됨에 따라 다양한 분야에서 인공지능을 활용하고 있다. 소스코드에 대한 취약점 분석 및 유사성 비교 분석을 위해서 기존 자연어 처리 기법과 더불어 이미지 처리기법을 활용한 연구가 활발하게 진행되고 있다[1].

특히 Malware 탐지 분야에서는 기존 소스코드 분석이나, 정적, 동적 탐지 기법이 아닌 코드를 이미지화 하여 인공지능 CNN 모델을 이용해 Malware 검출을 성공적으로 이룬 연구 결과들[2-6]이 발표되고 있다.

HoneyPot 사기와 Ponzi사기의 경우 스마트 컨트랙트의 사기 알고리즘이 스마트 컨트랙트 코드로 구현되어 있으며[2], 이는 일반적인 스마트 컨트랙트와는 다른 구성을 가지고 있다.

[†]E-mail: ybpark@dankook.ac.kr

본 논문은 malware 탐지에 성공한 CNN 모델을 동일하게 사용하여 HoneyPot 사기와 Ponzi 사기 알고리즘이 적용된 스마트 컨트랙트를 탐지해보고자 한다.

2. 관련 연구

2.1 Honeypot

설계에 취약점이 있는 스마트 컨트랙트로, 사용자가 자금 탈취를 위해 자금을 전송하면 생성자가 송금된 금액을 인출하는 사기이다. 이는 사용자가 취약점에만 초점을 두어 컨트랙트에 또 다른 취약점이 존재할 가능성을 고려하지 않기 때문에 발생한다.

2.2 Ponzi

높은 수익율을 미끼로 자금을 모은 후 신규 투자자의 자금으로 기존 투자자에게 이자나 배당금을 지급하는 다단계 사기이다. 정당한 수익 없이 투자자의 돈으로 다른

투자자들에게 수익을 제공하는 방식으로 투자자의 계약 가입 시점이 늦을수록 손실 위험이 커진다.

2.3 Visualized malware classification framework

Malware의 바이너리 값은 8비트 정수 벡터로 변경 한 후 이를 2차원 배열로 구성한 후 흑백 이미지로 변경한다 [5]. 생성된 이미지는 normal과 abnormal로 분류하는 신경망에 입력으로 들어가게 된다[6]. 시스템의 구조는 Fig. 1과 같다.

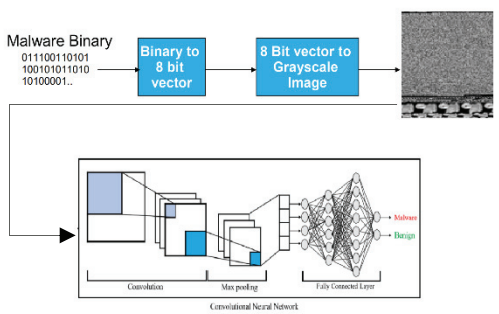

Fig. 1. Visualized malware classification framework.

3. 실험 방법

bytecode는 op 코드의 집합으로 스마트 컨트랙트를 컴파일 할 경우 얻을 수 있다. CNN에서 학습하기 위해서는 이를 이미지로 변경하는 과정을 필요로 한다.

따라서 Fig. 2과 같은 과정을 거쳐 이미지 데이터를 준비한다.

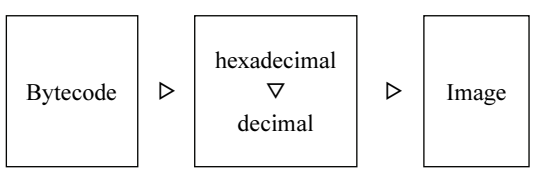

Fig. 2. Bytecode to Image.

3.1 Bytecode 수집

Normal 데이터의 경우 etherscan[7]에서 제공하는 Verified Contracts에서 1000개의 정상적인 컨트랙트 주소와 bytecode를 수집했다.

Abnormal는 [8]에서 제공하는 769개의 Honeypot 사기 데이터와 [9]에서 제공하는 174개의 Ponzi 사기 데이터를 합쳐 하나의 class로 구성하였다.

수집한 데이터는 컨트랙트 주소가 이름인 텍스트 파일로 만들어 각 class 종류에 맞는 폴더 안에 저장했다. 수집한 총 데이터의 수는 Table 1과 같다.

Table 1. Number of Files

Class	Number of files
Normal	1000
Abnormal	943

3.2 10 진수 변환

스마트 컨트랙트의 bytecode는 16진수로 되어 있다. 이미지 변경이 쉽게 하기위해서 16진수로 작성되어 있는 bytecode를 10진수로 변환한다. 이후 값차이를 극명하게 하기 위해서 변환된 10진수 값에 28을 곱하여 [0-255] 사이에 값이 들어올 수 있도록 만들었다.

3.3 이미지 데이터 생성

위와 같은 과정을 거쳐 데이터들은 1차원의 형상을 하고 있다. 하지만 이미지의 경우 2차원 값이 필요하므로 이를 2차원 배열로 변경해주는 과정을 거친다.

배열 생성 시 불필요한 데이터를 최소화하기 위해 표2와 같이 파일의 크기에 따라 배열의 너비를 다르게 지정한다. 이때 너비는 [2]에서 사용한 너비를 참고했다.

Table 2. Width by file size

File size	width
<=10kb	32
10kb~30kb	64
30kb~60kb	128
60kb<=	256

파일에 저장되어 있는 값은 전부 다르므로 파일의 크기마다 생성되는 배열의 높이는 모두 다르다. 너비 별 파일 개수는 Fig.3과 같다.

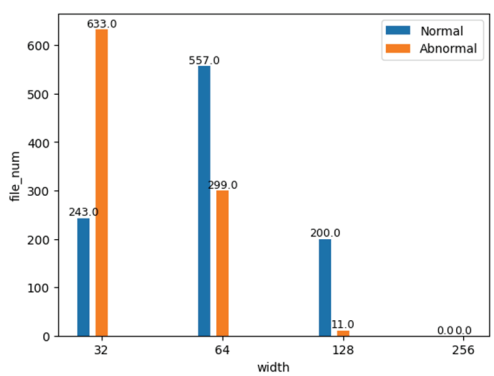

Fig. 3. 데이터 분포도.

2차원 배열로 변경된 데이터를 이용하여 파일 크기마다 너비와 높이가 다른 이미지를 생성한다. 이렇게 생성된 이미지 Fig. 4(a), Fig. 4(b)는 컬러 이미지이다.

Fig. 4. (a) Normal image.

Fig. 4. (b) Abnormal image.

3.4 모델 학습

여러 CNN 모델들을 이용하여 학습 통해 성능을 측정해 보았다. 평가한 모델들은 3 layer의 CNN, VGG16, VGG19, ResNet 50 그리고 ResNet 152로 총 5개이다. Figure 5(a), Figure 5(b), Figure 5(c), Figure 5(d), Figure 5(e)는 해당 모델들의 학습 결과를 보여주는 그래프이다.

VGG[10]는 다중 layer를 가지고 있는 CNN모델로, 3x3 필터를 사용한 convolution layer를 가지고 있다. VGG16는 convolution layer 모듈을 16개, VGG19는 19개를 가지고 있다.

Fig. 5(b), Fig. 5(c)는 각각 VGG16과 VGG19를 학습한 그래프이다. 이들은 학습 후반으로 갈수록 낮은 정확도에서 일정한 값을 그리며 학습이 제대로 이루어지지 않는 모습을 보였다.

ResNet[11]는 깊이가 깊어질수록 기울기 소실 문제가 발생하는 것을 해결하기 위해 residual learning을 사용한 CNN을 말한다. 뒤에 붙어있는 숫자는 해당 모델이 가지고 있는 convolution layer층을 말한다.

Fig. 5(d), Fig. 5(e)는 각각 ResNet50과 ResNet152를 학습한 그래프이다. 앞선 VGG와는 다르게 학습이 정상적으로 이루어지는 모습을 보였다.

하기의 Table 3은 각 모델의 측정된 validation acc를 표기하였다.

Table 3. Models compare results

Model	Validation acc (Highest)
3-layer CNN	0.8
VGG16	0.55
VGG19	0.55
ResNet 50	0.8
ResNet 152V2	0.81

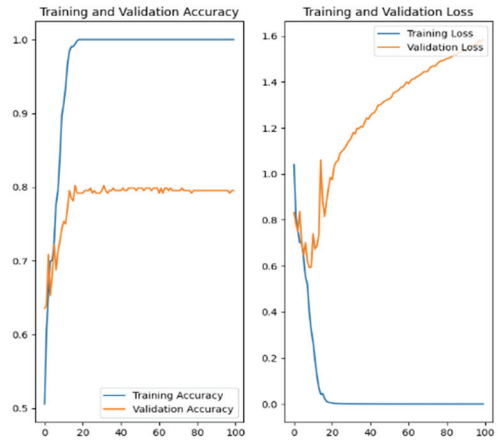

Fig. 5. (a) 3-layer CNN.

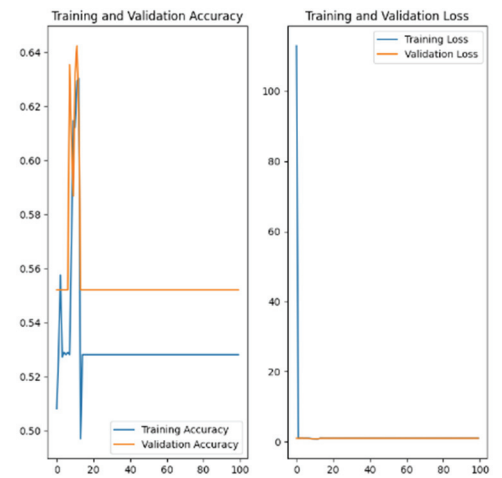

Fig. 5. (b) VGG 16.

Fig. 5. (c) VGG 19.

Fig. 5. (d) ResNet 50.

Fig. 5. (e) ResNet 152.

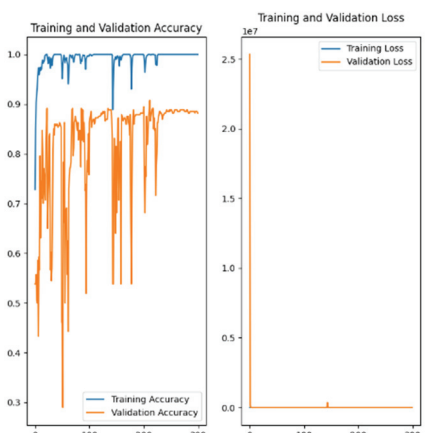

Fig. 5. (f) ResNet 152 – final.

4. 실험 결과

위 모델들 중 가장 학습이 잘 된 ResNet152 모델의 Percision, Recall 및 F1 score 평가를 진행했다.

평가를 위해 필요한 값은 TP(True Positive), FP(False Positive), FN(False Negative)이다. TP는 모델이 정답이라 예측한 것이 실제 정답인 경우를, FP는 모델이 정답이라 예측했지만 실제로는 오답인 경우를, 그리고 FN은 모델이 오답이라 예측한 것이 실제로 오답인 것을 말한다. 이것들을 이용하여 다음 평가 값들을 구한다.

Precision(정밀도)는 모델이 정답이라 예측한 것들 중 실제로 정답인 경우의 비율을 말한다. Precision 구하는 식은 (1)과 같다.

$$Percison = TP/(TP + FP) \quad (1)$$

Recall(재현율)은 실제로 정답일 때 모델이 정답이라 예측한 비율이다. Recall을 구하는 식은 (2)와 같다.

$$Recall = TP/(TP + FN) \quad (2)$$

F1 score은 정밀도와 재현율의 조화평균을 나타낸 지표로, class별 성능을 평가하며, 구하는 방법은 (3)과 같다.

$$f1\ Score = (2 * Precision * Recall)$$
$$(Precision + Recall) \quad (3)$$

위 모델들 중 가장 학습이 잘 된 ResNet152 모델의 Percision, Recall 및 F1 score 평가를 진행했다. 하기의 Table 4는 측정된 값이다.

Table 4. Evaluation results

	Value
Percision	0.825136612021858
Recall	0.893491124260355
F1 score	0.8579545454545454

측정결과 일반적인 이미지를 분류하는 CNN 모델들 같이 90이상의 성능이 나오지 않으나, 사기 등 비정상 데이터를 탐지하는데 중요한 지표인 Recall의 측정값이 90에 근접하여, 추후 보안을 통해서 충분히 탐지 모델로 활용이 가능함을 보인다.

5. 결 론

이미지 분석 기법을 통해서 소스코드의 유사성을 분석한 결과[1] 및 Malware 탐지[2-6]와 마찬가지로 스마트 컨트랙트 또한 bytecode를 이미지로 변경하여 CNN 모델을

통해서 비정상 논리를 가진 사기 스마트 컨트랙트 탐지가 가능하다.

여러 CNN 모델을 시험한 결과 컨볼루션 레이어가 적은 모델 보다는 레이어 수가 많은 모델일 수록 학습을 통해 비정상 데이터를 탐지해낼 가능성이 높다는 것을 확인했다.

가장 성능이 높았던 ResNet152V2의 경우 F1score 0.85를 달성하였다. 이후 보다 많은 학습데이터와 적절한 전처리를 통해서 더 높은 성능을 가진 모델을 개발할 수 있을 것으로 보인다.

감사의 글

이 논문은 2023년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (2021-0-00177, 스마트 컨트랙트의 개발-배포-실행의 전주기적 취약점 및 신뢰성 오류 개선 기술개발)

참고문헌

1. Dong-Bin Choi, In-su Jo and Young B. Park. (2021) Comparison of Code Similarity Analysis Performance of funcGNN and Siamese Network, Journal of the Semiconductor & Display Technology, 20(3), 113-116.
2. F. Zhong, Z. Chen, M. Xu, G. Zhang, D. Yu and X. Cheng, “Malware-on-the-Brain: Illuminating Malware Byte Codes With Images for Malware Classification,” in IEEE Transactions on Computers, vol. 72, no. 2, pp. 438-451, 1 Feb. 2023, doi: 10.1109/TC.2022.3160357.
3. Marastoni, Niccolò, Roberto Giacobazzi, and Mila Dalla Preda. “Data augmentation and transfer learning to classify malware images in a deep learning context.” Journal of Computer Virology and Hacking Techniques 17 (2021): 279-297.
4. Daoudi, Nadia, et al. “Dexray: a simple, yet effective deep learning approach to android malware detection based on image representation of bytecode.” Deployable Machine Learning for Security Defense: Second International Workshop, MLHat 2021, Virtual Event, August 15, 2021, Proceedings 2. Springer International Publishing, 2021.
5. Nataraj, Lakshmanan, et al. “Malware images: visualization and automatic classification.” Proceedings of the 8th international symposium on visualization for cyber security. 2011.
6. Pinhero, Anson, et al. “Malware detection employed by visualization and deep neural network.” Computers & Security 105 (2021): 102247.
7. “Verified Contracts”, Etherscan, accessed Aug 07, 2023, https://etherscan.io/contractsVerified
8. C. F. Torres, M. Steichen, and R. State, “The art of the SCAM: Demystifying honeypots in ethereum smart contracts,” in Proc. 28 USENIX Security Symp., 2019, pp. 1591–1607.
9. M. Bartoletti et al., “Dissecting ponzi schemes on ethereum: identification, analysis, and impact,” arXiv preprint arXiv:1703.03779, 2017.
10. Simonyan, Karen, and Andrew Zisserman. “Very deep convolutional networks for large-scale image recognition.” arXiv preprint arXiv:1409.1556, 2014.
11. He, Kaiming, et al. “Identity mappings in deep residual networks.” Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14. Springer International Publishing, 2016.

접수일: 2023년 8월 16일, 심사일: 2023년 9월 6일,
게재확정일: 2023년 9월 12일