머신러닝을 이용한 N-gram Opcode 기반 악성파일 탐지*

이유진,[1†] 임정수[2‡]

[1,2]서울여자대학교 (학부생)

N-gram Opcode based Malicious File Detection Using Machine Learning*

Yu-Jin Lee,[1†] Jeong-Su Lim[2‡]

[1,2]Seoul Women's University (Undergraduate Student)

요 약

나날이 발전하는 악성코드를 탐지하기 위해 머신러닝 기반 탐지 연구가 중요해지고 있다. 악성코드는 주로 특정 Opcode 시퀀스를 포함하므로 특정 파일에 악성 Opcode 시퀀스가 존재한다면 악성파일이라고 분류할 수 있다. 본 논문에서는 악성코드 실행파일로부터 추출한 Opcode 시퀀스를 N-gram 기반 전처리 방법인 Opcode Count, Opcode Frequency 및 TF-IDF을 사용하여 1-gram 및 2-gram을 기준으로 분석한다. 그리고 Gaussian Naive Bayes, Support Vector Machine, Random Forest, Decision Tree, K-Nearest Neighbor 머신러닝 모델을 사용하여 학습하고 정확도를 비교한다. 실험 결과 2-gram TF-IDF 방법으로 전처리하여 Random Forest 모델로 학습하였을 때 가장 높은 정확도를 달성하였다.

I. 서론

1986년 최초의 컴퓨터 바이러스 출현 이후 끝없이 새로운 바이러스들이 발견되고 있다. AV-Test Institute에서 발표한 자료에서는 하루 평균 450,000개의 새로운 악성코드가 탐지되고 있다고 한다. 특히, 코로나19 이후 악성코드에 감염된 사이트는 더욱더 증가하였으며 [1], 재택근무를 하는 기업들을 노려 해당 기업의 네트워크에 불법으로 침입하여 유포하는 등, 악성코드는 여러 경로를 통해 기업과 개인을 위협하고 있다. 또한, 머신러닝과 결합된 신종 악성코드가 늘어남에 따라 더 이상 과거처럼 악성코드 분석가의 수작업 분석에 의존하는 것이 어려워졌다. 이러한 한계를 극복하기 위하여 머신러닝을 기반으로 악성코드를 탐지하고 분석하고, 나아가 예방하는 연구의 필요성이 더욱 대두되고 있다.

악성코드 실행파일에서 추출된 Opcode 시퀀스를 기반으로 악성코드를 분석하면 로우 데이터에서 피처를 학습할 수 있다는 장점이 있다. Opcode 시퀀스로부터 학습 입력 데이터를 생성하는 기술로 N-gram 기법이 있다. N-gram은 많은 머신러닝 기반 악성코드 연구에서 특징으로 사용되었고, 특히 정적 분석에 있어 가장 일반적으로 사용되는 특징 유형 중 하나다 [2].

본 연구에서는 N-gram Opcode 기반 전처리 방법인 Opcode Count, Opcode Frequency, TF-IDF(Term Frequency-Inverse Document Frequency)를 사용하여 1-gram 및 2-gram을 기준으로 분석하고, 어떤 전처리 모델이 악성코드를 분류하는 데 가장 좋은 방법인지 실험한다. 또한, Gaussian Naive Bayes, Support

* 본 연구는 서울여자대학교 SW중심대학추진사업단 지원의 연구결과로 수행되었음 (2022).

† 주저자, pqk@swu.ac.kr

‡ 주저자, jamielim503@swu.ac.kr

Vector Machine (linear kernel), Support Vector Machine (RBF kernel), Random Forest, Decision Tree, K-Nearest Neighbor의 6개 머신러닝 모델을 사용하여 Opcode의 N-gram을 적용한 전처리 방식의 악성파일 탐지에 있어 어떤 모델이 가장 좋은 성능을 보이는지 비교 및 분석한다.

II. 관련 연구

Mohandas 등 [3]은 악성파일을 식별하기 위해 파일을 어셈블리 언어로 변환한 후 Opcode를 추출하였다. 이때, Opcode Count 방식과 Frequency 방식을 사용해 분류 모델에 전달한 후 수신 파일에 잠재적인 악성 코드가 포함되어 있는지 여부를 예측하는 연구를 제안하였다.

Shabtai 등 [4]은 Opcode의 n-gram 패턴을 사용해 악성과 정상파일을 분류하였다. 해당 연구에서는 n-gram 패턴에 각각 normalized term frequency (TF)와 TF inverse document frequency (TFIDF) 방식을 적용하여 분류 모델의 정확도를 측정하였다. 그 결과 6-gram의 TF 방식이 95.375%로 가장 높은 정확도를 기록하였다.

Kim 등 [5]은 악성코드 탐지 기법의 무력화를 방지하기 위해 악성코드에 적용될 수 있는 패커의 특성을 활용해 분석 방지 보호 기법을 탐지 및 분류하는 머신러닝 모델을 구축하였다. 해당 연구에서는 N-gram Opcode를 추출한 뒤 TF-IDF를 활용하여 피처를 추출하였다. 이를 통해 6개의 머신러닝 모델로 실험한 결과 Themdia에서 81.25%의 정확도를 VMProtect에서 95.65%의 정확도를 기록하였다.

III. N-gram Opcode 기반 악성파일 탐지 방법

3.1 데이터 전처리

본 연구에서는 N-gram Opcode 기반으로 정적 분석을 통해 악성코드 분석을 하여 악성파일을 탐지한다. 악성 실행 파일의 실행 섹션에서 각 파일의 opcode 시퀀스를 추출하였다. 하나의 opcode는 1바이트를 사용해서 표현하였다. 하나의 opcode 시퀀스는 각 코드 라인별로 Address, Hex Opcode, Opcode, Operand로 파싱하여 리스트로 구성한 뒤, CSV 형태로 변환하여 분석에 사용하기 용이하게 한다. 그리고 추출한 opcode 시퀀스를 연속된 n개의 바이트로 그룹화하여 워드를 생성한다. 본 연구에서는 1-gram 및 2-gram을 사용하였다.

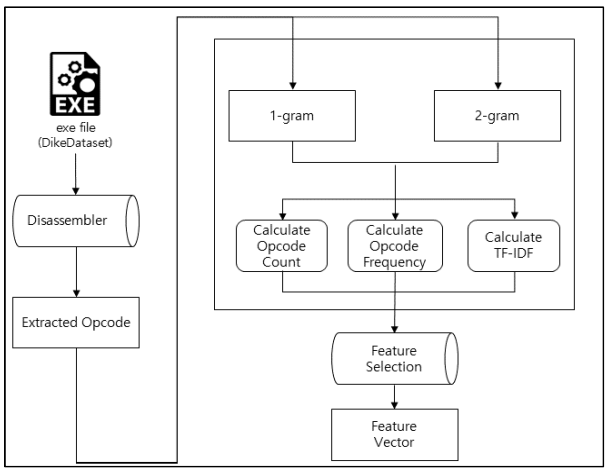

Fig. 1. Overview of Feature Selection

3.2. Opcode Count를 이용한 탐지 방법

$$Opcode\ \ Count = No.\ of\ Instances\ of\ the\ Opcode \quad (1)$$

3.1절에서 전처리를 마친 N-gram Opcode 시퀀스를 담은 각 파일에서 Opcode Count를 계산한다. Opcode Count는 특정 Opcode가 등장한 횟수이다.

3.3 Opcode Frequency를 이용한 탐지 방법

$$Opcode\ \ Frequency = \frac{No.\ of\ Instances\ of\ the\ Opcode}{Total\ No.\ of\ Opcodes\ in\ the\ files} \quad (2)$$

Opcode Frequency는 수식 2와 같이 계산한다 [3]. 파일에서 특정 Opcode가 발생한 횟수를 파일에서 Opcode가 발생한 총 횟수로 나눈다.

이때 파일에서 특정 Opcode가 발생한 횟수는 Opcode Count와 동일한 의미이다.

3.4 TF-IDF를 이용한 탐지 방법

$$tf_{t,d} = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \quad (3)$$

$$idf_{t,D} = log \frac{|D|}{|d \in D : t \in d|} \quad (4)$$

$$tfidf_{t,d,D} = tf(t,d) \cdot idf(t,D) \quad (5)$$

3.1절에서 구한 N-gram Opcode는 각 파일 내에서 다양한 종류와 많은 개수로 이루어져 있다. 따라서 N-gram Opcode 중에서 의미 있는 N-gram Opcode를 추출하는 방법을 시도할 수 있다. 본 논문에서는 TF-IDF를 사용하여 파일 집합 전체에 대한 특정 N-gram Opcode의 중요성을 분석하였다. TF(Term Frequency)는 수식 3과 같이 계산하며, 특정 파일 d에서 주어진 N-gram Opcode t의 빈도를 나타낸다. 여러 파일에서 동일하게 자주 나타나는 N-gram Opcode는 특정 악성파일에서만 나타나는 특징으로 볼 수 없으므로, 유의미하지 않은 N-gram Opcode라고 할 수 있다. 따라서, TF를 역으로 계산한 IDF(Inverse Document Frequency)를 계산하여, 보편적으로 등장하는 N-gram Opcode를 제외했으며, 이에 대한 계산은 수식 5에 있다.

수식 5에서 $|D|$는 전체 파일의 수를 의미한다. 파일이 늘어날수록 idf의 값이 무한히 증가하는 것을 방지하기 위하여 log를 취한다. TF-IDF는 수식 4처럼 계산한다. tf에 idf를 곱한 값을 3.1절에서 구한 파일마다 계산한다.

IV. 악성파일 탐지 모델 구현 및 실험

4.1 실험 대상

실험 대상으로 DikeDataset로부터 APT1, APT10, APT19, APT21, APT28, APT29, APT30, Dark Hotel, Energetic Bear, Gorgon Group, WinNTI 등에 속하는 일반적인 악성코드 샘플 10,841개를 수집하였다. 각 악성 실행파일의 실행 영역을 디스어셈블 하여 opcode를 추출하였다. 각 파일에 고유한 해시를 생성하여 중복을 제거하고, 이상치를 제거하였다. 그 결과 4,223개의 악성파일과 938개의 정상파일을 얻었다. 악성파일 8,938개 중 6,704개는 training 데이터로, 2234개는 test 데이터로 구성하였고, 정상파일은 총 949개 중 703개는 training 데이터로, 234개는 test 데이터로 구성하였다. 이 데이터셋을 각각 1-gram 및 2-gram으로 BOW(Bag Of Wards) 인코딩 벡터를 생성하여서 Opcode Count 데이터셋을 구축하였고, 3.3절에서 제시한 식을 적용하여 Opcode Frequency 데이터셋을 구축하였다. 3.4절에서 제시한 식을 적용하여 TF-IDF 데이터셋을 구축하였다.

4.2 실험 결과 및 분석

6개의 머신러닝 모델을 4.1절에서 제시한 1-gram 및 2-gram을 적용한 Opcode Count(OC), Opcode Frequency(OF), TF-IDF 각각으로 전처리 된 데이터셋으로 학습시키고 평가하여 가장 좋은 성능의 악성파일 분류 방식을 비교하였다.

Table 1. Comparison of Model Detection Accuracy (%)

Model	1-gram			2-gram		
	OC	OF	TF-IDF	OC	OF	TF-IDF
NB	92.48	93.10	74.28	66.07	83.73	84.50
SVM (linear)	97.75	98.99	96.04	99.22	99.61	99.30
SVM (RBF)	90.31	98.14	98.14	93.88	99.38	99.30
RF	99.38	99.38	99.30	99.45	99.45	99.53
DT	97.52	98.21	98.14	98.91	99.14	98.76
KNN	97.67	98.21	98.37	97.98	98.68	98.60

전반적으로 Random Forest 모델이 높은 정확도를 보였다. 여러 모델의 결과를 비교하여

더 좋은 성능을 내는 앙상블 학습 기법이 N-gram Opcode를 기반으로 악성파일을 탐지하는 데에 좋은 성능을 보인다는 것을 알 수 있다. 특히, 2-gram에 TF-IDF를 적용한 전처리 방식이 99.53%의 높은 수치를 기록하였다. 반면 Gaussian Naive Bayes 모델이 가장 낮은 정확도를 기록한 것을 확인할 수 있다. Naive Bayes 모델은 데이터가 서로 독립적임을 가정하는데, 본 논문에서 사용한 N-gram Opcode는 서로 상당히 의존적이다. 이러한 이유로 Naive Bayes 모델은 본 논문에 적합하지 않다는 것을 알 수 있다.

N-gram Opcode에 1-gram을 적용한 경우, Opcode Frequency 전처리 방식이 K-Nearest Neighbor 모델의 경우를 제외하고는 전체적으로 높은 정확도를 기록한 것을 확인할 수 있다. N-gram Opcode에 2-gram을 적용한 경우에도 대부분의 모델에서 Opcode Frequency를 사용한 경우에 높은 정확도를 보였다.

반면, Opcode Count는 대부분의 1-gram과 2-gram 적용 모두 모델에서 가장 낮은 정확도를 보여주였다. 특히, 1-gram을 적용한 RBF를 사용하는 Support Vector Machine 모델과 2-gram을 적용한 Gaussian Naive Bayes 모델에서는 다른 전처리 방식과 각각 약 8%, 약 18%의 큰 정확도 차이를 보였다.

V. 결론

본 논문에서는 실제 악성 및 정상파일에서 추출한 N-gram Opcode를 기반으로 여러 전처리 방법을 통해 악성파일을 탐지하고 결과를 비교하였다. 이때 Opcode Count, Opcode Frequency, TF-IDF을 1-gram 및 2-gram을 기준으로 각각 적용하여 전처리하였으며, 이를 6가지의 머신러닝 모델에 학습시키고 정확도를 비교하여 가장 정확하게 악성파일을 탐지하는 모델을 비교 및 분석하였다.

그 결과 Opcode Frequency로 전처리하였을 때 1-gram 및 2-gram 전체에서 높은 정확도를 기록하였고, 머신러닝 모델의 경우 Random Forest 모델이 다른 모델에 비해 높은 정확도를 보였다. 특히, 2-gram에 TF-IDF를 적용한 전처리 방식이 99.53%의 높은 수치를 기록하였다.

2-gram 이상의 경우 및 다른 머신러닝을 사용한 경우에도 동일한 결과를 도출하는지에 대하여 연구하는 것이 향후 과제이다.

[참고문헌]

[1] Malware Statistics & Trends Report | AV-TEST . (n.d.).

[2] Kang, B., Yerima, S. Y., Sezer, S., & McLaughlin, K. (2016). N-gram Opcode analysis for android malware detection. arXiv preprint arXiv:1612.01445.

[3] Mohandas, P., Kumar, S. K. S., Kulyadi, S. P., Raman, M. S., Vasan, V. S., & Venkataswami, B. (2021, July). Detection of malware using machine learning based on operation code frequency. In 2021 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT) (pp. 214-220). IEEE.

[4] Shabtai, A., Moskovitch, R., Feher, C., Dolev, S., & Elovici, Y. (2012). Detecting unknown malicious code by applying classification techniques on opcode patterns. Security Informatics, 1(1), 1-22.

[5] Canfora, G., Mercaldo, F., & Visaggio, C. A. (2015, July). Mobile malware detection using op-code frequency histograms. In 2015 12th International Joint Conference on e-Business and Telecommunications (ICETE) (Vol. 4, pp. 27-38). IEEE.

[6] 김희연, & 이동훈. (2022). N-gram Opcode를 활용한 머신러닝 기반의 분석 방지 보호기법 탐지 방안 연구. 정보보호학회논문지, 32(2), 181-192.