정적 분석과 MITRE ATT&CK TTPs 행위 분석 데이터를 활용한 머신러닝 기반 악성코드 탐지 방안

안도현[O1] 설기현[2] 진소정[3] 이성철[4] 이상준[5] 김현민[6] 김두민[7] 손승호[8]
[1]고려대학교 세종캠퍼스 인공지능사이버보안학과
[2]서울과학기술대학교 컴퓨터공학과
[3]아주대학교 정보통신공학과(석사과정)
[4]광주대학교 사이버보안경찰학과
[5]건국대학교 글로컬캠퍼스 ICT융합공학부
[6]금융보안원
[7]SK텔레콤
[8]고려대학교 소프트웨어보안연구소
dksehgus1102@korea.ac.kr, seolpark@seoultech.ac.kr, jinso96@ajou.ac.kr, lee41917@gmail.com,
scottlee199909@gmail.com, hyunmini85@gmail.com, kdm820@naver.com, heap@kakao.com

Machine Learning-based Malware Detection Using MITRE ATT&CK TTPs Derived Features from Static Analysis

Do-Hyun Ahn[O1] Ki-Hyeon Selo[2] So-Jeong Jin[3] Seong-Cheol Lee[4]
Sang-Jun Lee[5] Hyun-Min Kim[6] Doo-Min Kim[7] Seung-Ho Son[8]
[1]Dept, of AI Cyber Security, Korea University Sejong Campus
[2]Dept, of Computer Science and Engineering, Seoul National University of Science and Technology
[3]Dept, of Information and Communication Technology, Ajou University(Graduate Student)
[4]Dept, of Cyber Security and Police, Gwangju University
[5]Dept, of Information and Cmmunication Technology Convergence Engineering, Konkuk University Glocal Campus
[6]Financial Security Institute
[7]SK Telecom
[8]Center for Software Security & Assurance, Korea University

요 약

최근 우크라이나 사태 관련 사이버전 확산, 국내 기업 대상 랜섬웨어, 정보 유출 사고 등 국내외 사이버 위협 발생 가능성이 고조되고 있다. 사이버 위협 동향에 따르면 공격자들은 다양한 경로와 공격 방법을 통해 악성코드를 유포하고 있으며, 악성코드를 활용하여 사이버 공격 활동을 진행하고 있다. 기존에 악성코드를 탐지하는 방식 중 하나인 시그니처 방식은 정해진 패턴에 기반하여 악성코드를 탐지하기 때문에 발견되지 않은 악성코드를 탐지하는데 있어 어려움을 겪고 있다. 따라서 본 논문에서는 CAPA 도구를 활용하여 머신러닝 학습에 필요한 피처를 추출하고 데이터 세트를 구성해 알려지지 않은 악성코드에 대해서도 탐지할 수 있는 최적의 머신러닝 모델 도출 방안에 대해 제안한다.

1. 서 론

악성코드는 그 존재만으로도 컴퓨터 시스템에 심각한 위협을 불러올 수 있으며, 이를 대표하는 바이러스, 웜, 트로이 목마, 애드웨어, 스파이웨어, 랜섬웨어 등은 사용자의 정보를 탈취하거나 시스템을 마비시키는 등의 행위를 목적으로 한다. 이러한 악성코드의 활동을 미리 감지하고 대응하기 위한 '악성코드 탐지'는 컴퓨터 시스템을 손상하거나 사용 불가능 상태로 만드는 악성 소프트웨어의 유해한 영향을 식별, 차단 및 방지하는 필수적인 기술이다. 현대의 다양한 악성코드 유형과 지속적으로 발전하는 테크닉에 대응하기 위해, 고도화된 탐지 기법의 연구와 개발이 절실하게 요구되고 있다.

ITU(International Telecommunication Union)의 최근 통계에 따르면, 전 세계 인구의 약 67%가 인터넷을 사용하는 것으로 추정된다. 이는 2018년에 비해 압도적인 45%의 증가를 보인 수치로, 디지털 연결성의 급속한 확대와 정보화 시대의 도래를 뚜렷하게 보여준다[1]. 이러한 확대된 인터넷 사용률은 개인과 조직에게 많은 이점을 제공하나, 동시에 온라인 환경에서의 위협, 특히 악성코드와 같은 보안 위협에 노출될 가능성도 크게 증가시킨다. 따라서 인터넷 사용자의 증가와 함께, 악성코드에 대한 방어 기술의 중요성도 상승하게 되는 상황이다.

기존에 널리 사용되어 온 백신과 같은 시그니처 기반 악성코드 탐지 솔루션은 정해진 패턴에 기반하여 악성코드를 탐지하였다. 그러나, 악성코드 제작자들은 지속적으로 새로운 기법과 변형을 통해 탐지를 회피하려 하며, 특히 제로데이 악성코드와 같은 미리 알려지지 않은 공격 기법은 시그니처 기반의 탐지 방법만으로는 감지가 어렵다는 한계를 보이게 되었다. 이러한 변화의 바람에, 최근에는 악성코드의 탐지 방식에 혁신을 주려는 시도가 활발히 이루어지고 있다. 그중에서도 기계 학습을 활용한 악성코드 탐지는 그 능력으로 많은 주목을 받고 있다. 기계 학습은 악성코드의 다양한 행동과 패턴을 학습하여, 시그니처가 없는 새로운 악성코드도 높은 정확도로 탐지하는 능력을 갖추게 되었다. 이런 배경으로, 기계 학습과 악성코드 탐지의 결합에 대한 연구와 개발이 급속도로 진행되고 있는 현재 상황이다[2, 3].

기계 학습의 워크플로우는 크게 다섯 단계로 구분될 수 있다.

(1) **수집 단계**: 이 단계에서는 특정 목적 또는 문제를 해결하기 위해 필요한 데이터를 수집한다. 이 데이터는 공개 데이터셋, 사용자의 행동 로그, 센서에서 발생하는 신호 등 다양한 출처에서 수집될 수 있다.

(2) **데이터 전처리 단계**: 수집된 데이터는 종종 누락된 값, 이상치, 잡음 등의 문제를 포함할 수 있다. 이러한 데이터를 알고리즘에 직접 사용하면 성능이 저하될 수 있기 때문에,

데이터 전처리를 통해 이를 깨끗하고 일관된 형태로 변환한다.

(3) **특징 공학 단계**: 이 단계에서는 도메인 지식을 활용하여 데이터에서 중요한 특징을 추출하거나 변환한다. 올바른 특징을 선택하고 설계하는 것은 모델의 성능에 결정적인 영향을 미친다.

(4) **모델링 단계**: 적절한 알고리즘을 선택한 후, 전처리된 데이터와 특징들을 사용하여 모델을 학습시킨다. 여기서 선택되는 알고리즘은 문제의 종류, 예를 들면 회귀, 분류, 클러스터링 등, 및 데이터의 특성에 따라 다르다.

(5) **모델 튜닝 단계**: 초기에 학습된 모델은 최적의 성능을 보장하지 않을 수 있다. 따라서 이 단계에서는 모델의 성능을 평가하고, 필요한 경우 파라미터를 조정하여 성능을 개선한다. 이 과정은 여러 번 반복될 수 있으며, 최적의 성능을 달성하기 위해 다양한 튜닝 기법이 사용된다.

정적 분석을 통한 특징 추출은 소프트웨어를 실행시키지 않고도 코드 및 파일의 구조 등을 파악하여 내부에 포함된 함수 호출 패턴, API 사용법, opcode 연속성 그리고 바이트 시퀀스와 같은 다양한 정보를 찾아낸다[4].

본 논문에서는 다양한 바이너리 코드의 패턴, API 호출 및 문자열과 같은 정보를 추출하기 위한 기준으로 CAPA-rule을 활용한다. 이를 통해, PE 파일의 내부 구조와 기능을 세밀하게 파악할 수 있게 된다. 더 나아가, 이러한 정보는 악성코드의 전술과 기술에 대한 표준화된 규칙을 제공하는 MITRE ATT&CK 프레임워크의 Techniques, MBC(Malware Behavior Catalog)와 연계되어, 악성코드의 특징을 더욱 구체적이며 체계적으로 기술하게 된다. 이렇게 추출된 특징들은 기계 학습 모델의 학습 데이터로 활용되며, 본 논문에서는 이를 기반으로 한 학습 및 모델의 성능 평가 과정을 상세히 소개한다.

2. 머신러닝 학습 데이터 세트 구축

본 연구는 악성코드의 분류 문제를 깊게 탐구하기 위해 다양한 데이터 세트를 사용하여 학습 및 테스트를 진행하였다. 특히, 한국인터넷진흥원(KISA) 사이버보안 빅데이터센터에서 제공받은 데이터 세트와 합쳐 총 102,056개의 데이터 세트를 활용하였다(표 1). 이 중 KISA1 데이터 세트는 2017년 K-시큐리티 챌린지 악성코드 탐지 트랙에 사용된 데이터이며, KISA2와 KISA3은 각각 K-시큐리티 챌린지 2018, 2019년에 사용된 데이터이다.

사용된 데이터 세트는 정상 데이터 43,325개, 악성 데이터는 58,731개로 분포하고 있다. 이러한 데이터들의 형식과 특징을 살펴보면, 바이너리 형식은 대부분이 PE(86,448개)로 이루어져 있으나 dotnet 형식도 15,608개 포함되어 있다. 또한, 이 바이너리들은 시스템 운영 환경에 따라 32비트는 86,503개, 64비트는 15,553개로 분류되며, 이는 다양한 환경에서의 악성코드 행동을 포괄적으로 분석하고자 하는 연구의 의도를 반영한다.

3. 데이터 전처리 및 피처 엔지니어링

3.1 피처 추출(Feature Extraction)

본 연구에서는 악성코드의 본질적인 특징을 깊게 이해하고 이를 효과적으로 추출하기 위해 CAPA라는 도구를 사용한다.

표 1 학습에 사용된 데이터 세트 수집 현황

	정상	악성	계
KISA1	8,374	21,302	29,676
KISA2	5,068	12,585	17,653
KISA3	8,079	13,152	21,231
기타	21,804	11,692	33,496
계	43,325	58,731	102,056

CAPA는 PE 파일 내의 바이너리 패턴을 분석하여, 그 안에 내재된 다양한 기능과 행위를 식별하는 데 있어 독보적인 성능을 발휘한다. 중요한 점은, CAPA가 이러한 패턴 분석을 CAPA-rules를 통해 시행하며, 이러한 규칙들이 이미 MITRE ATT&CK technique과 MBC와의 연계성을 가지고 있다는 것이다.

이러한 연계성은 악성코드 분석에 있어서 중요한 의미를 가진다. 예를 들어, 'create reverse shell'이라는 규칙은 특정 API 조합을 통해 reverse shell 행위를 식별한다. 이러한 판단 기반이 되는 API 조합들은 특정 행위나 기능의 존재 여부를 확실히 하는 데 중요한 역할을 한다. 또한, 해당 규칙은 ATT&CK과 MBC에 정의된 특정 행위나 기능과 직접 연결되어 있다. 이렇게 함으로써, CAPA[5]를 통한 분석 결과는 보안 연구자나 전문가가 이해하기 쉬운 형태로 제시되며, 실제 위협 시나리오와 연계하여 악성코드의 행위를 구체적으로 파악할 수 있다.

CAPA v6.1.0을 사용한 본 연구에서는 833개의 기능을 성공적으로 매칭하였고, 이러한 기능들은 ATT&CK의 71개 테크닉과 MBC의 259개 항목과 연계되어 있음을 확인하였다. 이를 통해 본 연구는 악성코드의 깊은 통찰력을 제공하며, 보안 전문가들이 이를 기반으로 효과적인 대응 전략을 수립하는 데 도움을 줄 것이다.

3.2 피처 변환(Feature Transform)

머신러닝 모델의 학습과정에서는 데이터의 전처리와 스케일링이 성능 향상에 결정적인 역할을 한다. 특히, 여러 종류의 데이터를 결합할 때 이러한 과정은 더욱 중요해진다.

본 연구에서 CAPA를 통해 추출된 MITRE ATT&CK과 MBC 정보는 악성코드의 동작이나 특성을 범주형으로 표현한 것이다. 그 반면 capability는 악성코드가 수행하는 구체적인 기능의 수량을 나타내기 때문에 이산형 데이터의 형태를 갖는다. 따라서, 이 두 형태의 데이터를 모델에 함께 입력하기 전에 동일한 스케일에 맞추는 작업이 필요하다.

범주형 데이터인 MITRE ATT&CK과 MBC는 원-핫 인코딩 방식을 통해 수치형 데이터로 변환되었다. 원-핫 인코딩은 각 범주에 대해 유니크한 이진 값(0 또는 1)을 할당하는 방식으로, 머신러닝 모델이 이해하고 처리할 수 있는 형태로 데이터를 변환하는 효과적인 방법이다.

3.3 데이터 전처리(Data PreProcessing)

capability는 이미 정수형 값을 갖지만, 그 값의 범위가 크기 때문에 Robust Scaling 방식을 활용하여 이를 정규화하였다. Robust Scaling은 데이터의 중앙값과 IQR(Interquartile Range)을 사용하여 이상치의 영향을 최소화하면서 데이터를 스케일링하는 방법이다.

추가로, 파일의 FORMAT 정보와 ARCH 정보도 범주형 데이터로서 원-핫 인코딩이 적용되었다. 이를 통해 모든 입력 데이터가 동일한 스케일로 변환되어, 머신러닝 모델의 학습 과정에서 최적의 성능을 발휘할 수 있도록 되었다.

3.4 피처 선택(Feature Selection)

기계 학습 모델은 데이터의 차원이 클수록 그 복잡성도 증가한다. 복잡성이 증가하게 되면 모델이 학습 데이터에 과도하게 적응하려는 경향이 있어, 새로운 데이터에 대한 예측 성능이 저하될 수 있다. 이를 과적합이라고 부르며, 이는 모델이 너무 세세한 정보나 노이즈까지 학습하게 되어 일반화 능력을 잃어버리는 상황을 의미한다. 또한, 차원이 높아질수록 필요한 연산량도 증가하게 되어 학습과 검증에 많은 시간과 자원이 소모된다.

CAPA 도구를 이용하여 추출된 특징들 중 일부는 데이터 전체에서 존재하지 않는다. 이러한 특징들은 학습 과정에서 별다른 정보를 제공하지 않으므로, 모델의 성능 향상에 기여하지 않을 뿐만 아니라, 오히려 모델의 성능을 저하시킨다.

표 2 머신러닝 알고리즘별 학습 결과 및 수치 데이터

Model	Accuracy	Class 1 Precision	Class 1 Recall	Class 1 F1-Score
XGBoost	0.925	0.932	0.940	0.936
RandomForest	0.941	0.955	0.944	0.949
Gradient Boosting	0.874	0.881	0.906	0.893
AdaBoost	0.852	0.865	0.884	0.874
Extra Trees	0.939	0.927	0.940	0.947
LightGBM	0.915	0.924	0.932	0.928
CatBoost	0.926	0.935	0.939	0.937
Decision Tree	0.922	0.935	0.931	0.933

따라서, 이 연구에서는 효율적인 학습을 위해 사용되지 않는 특징들을 제거하여 차원을 축소하는 접근 방식을 선택했다. 이를 통해 모델의 학습 속도와 일반화 능력을 향상시키는 동시에, 과적합과 같은 문제를 방지하려는 노력을 기울였다.

4. 머신러닝 알고리즘 학습을 통한 모델링

4.1 머신러닝 알고리즘 학습 과정

본 연구에서는 Decision Tree, Random Forest, Extra Tree, XGBoost, GradientBoost, AdaBoost, LightGBM, CatBoost와 같은 8개의 알고리즘들을 통해 학습을 진행하였다.

학습 과정에서 선택한 8개의 머신러닝 모델은 안정성과 과적합의 여부를 검증하기 위해 5-fold 교차검증 방법을 적용하였다. 교차검증은 모델의 성능을 보다 일반적으로 평가하는 데 유용하며, 과적합의 위험을 줄이는 효과가 있다. 이때, 전체 데이터 세트의 80%를 학습에 사용하고, 나머지 20%는 테스트를 위해 보관하였다. 학습 데이터의 일관성을 유지하기 위해, 데이터 분할 시 랜덤 시드 값인 random_state를 42로 설정했다. 이러한 접근 방식은 학습 데이터의 무작위성을 줄이며, 모델 간의 성능 비교 시 불필요한 변동성을 최소화하는 데 도움을 준다.

4.2 머신러닝 알고리즘 유형

Random Forest와 Extra Tree는 앙상블 학습 방법 중 배깅(bagging)을 활용하는 모델이다. 이 두 모델은 본 연구에서 높은 성능을 보여주었다. 배깅 방식은 복원 추출 방법으로 여러 개의 표본 데이터 셋을 생성하고, 이를 바탕으로 각각의 결정 트리를 학습시킨다. 여러 트리들의 예측 결과를 평균내거나 다수결 투표 방식으로 최종 결과를 도출함으로써, 단일 결정 트리의 과적합 문제나 높은 변동성을 줄이는 효과가 있다.

부스팅(boosting) 계열의 앙상블 학습은, 학습 과정에서 발생하는 오차를 줄이기 위해 이전 모델의 오류를 보완하는 새로운 모델을 순차적으로 추가하는 방식이다. 이 연구에서는 LightGBM과 CatBoost가 부스팅 계열 모델 중 높은 성능을 나타냈다. LightGBM은 큰 데이터셋에 대한 빠른 학습 속도와 효율성으로 알려져 있으며, CatBoost는 범주형 특징의 자동 인코딩 기능으로 높은 성능을 보장한다.

4.3 트리기반 알고리즘의 특성

Decision Tree 기반의 앙상블 알고리즘은 여러 개의 나무를 조합하여 학습하는 특성 덕분에 개별 특징들이 예측 결과에 얼마나 크게 기여했는지를 수치적으로 평가할 수 있는 특징 중요도를 제공한다. 이러한 특징 중요도는 각 특징이 데이터 분할에 얼마나 자주 사용되었는지, 그리고 그 분할이 모델의 예측 성능 향상에 얼마나 기여하였는지를 기반으로 계산된다.

본 연구에서 가장 높은 성능을 보인 Random Forest 모델의 특징 중요도를 분석한 결과(그림 1), 'ARCH', 'contain loop', 'contains pdb path'가 상위에 위치했다는 것은 이들 특징이 악성코드의 분류에 있어 큰 역할을 했다는 것을 의미한다.

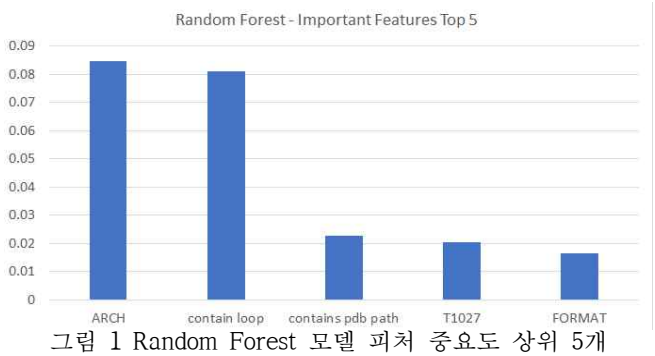

그림 1 Random Forest 모델 피처 중요도 상위 5개

5. 결론

본 연구에서는 CAPA 도구를 활용하여 악성코드로부터 머신러닝에 활용할 수 있는 특징을 추출하였다. 추출된 특징은 MITRE ATT&CK과 MBC에 매핑되어 분류되며, 이는 다양한 기능과 연관된 정보를 제공한다. 학습 데이터는 국내외 여러 데이터 세트를 포함하며, 데이터 세트는 바이너리의 형태, 아키텍처, 기타 특징에 따라 구분되었다.

데이터 전처리 과정에서 원-핫 인코딩과 Robust Scaling을 통해 머신러닝 알고리즘에 적합한 형태로 변환하였다. 여러 머신러닝 알고리즘을 활용하여 학습과 평가를 진행하였으며, 특히 Random Forest와 Extra Tree 모델이 높은 성능을 보였다(표 2).

Decision Tree 기반의 앙상블 알고리즘은 특징 중요도를 통해 학습 데이터의 각 특징이 예측에 얼마나 기여했는지 평가할 수 있었고, 본 연구에서 Random Forest 모델의 중요도 분석 결과 'ARCH', 'contain loop', 'contains pdb path' 등의 특징이 큰 영향을 미쳤음을 확인할 수 있었다.

이러한 결과를 통해 악성코드 분류에 있어서의 특징 추출과 선택, 그리고 적절한 머신러닝 알고리즘의 활용의 중요성을 확인하였다.

향후 연구에서는 다양한 특징과 머신러닝 알고리즘의 조합을 통해 더욱 높은 분류 성능을 달성하는 방안을 탐구할 예정이다. 또한 연구 결과 내용을 기반으로 악성코드로 인한 보안 위협이나 공격 확산 방지를 위해 행위 상태에 따라 단계를 격상시키며 네트워크 격리 및 감시를 통해 위협에 대한 신속한 상황 인식 및 대응 활동 징후를 탐지하고 효과적으로 대응할 수 있는 프레임워크 아키텍처를 제안할 예정이다.

참고 문헌

[1] International Telecommunication Union, **"ITU Statistics,"** [Online]. Available: https://www.itu.int/en/ITU-D/Statistics/Pages/stat/default.aspx.

[2] D. Gibert, C. Mateu, J. Planes, **"The rise of machine learning for detection and classification of malware: Research developments, trends and challenges,"** *Journal of Network and Computer Applications,* vol. 153, 2020, 102526.

[3] 최선오, 김영수, 김종현, and 김익균, **"딥러닝을 이용한 악성코드탐지 연구동향,"** *정보보호학회지,* vol. 27, no. 3, pp. 20-26, 2017.

[4] J. Singh and J. Singh, **"A survey on machine learning-based malware detection in executable files,"** *Journal of Systems Architecture,* vol. 112, 2021, 101861.

[5] Mandiant. **"CAPA."** 2023. [Online]. Available: https://github.com/mandiant/capa.