JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 31 No. 11, pp. 273-281, November 2026
https://doi.org/10.9708/jksci.2026.31.11.273

Hybrid Ensemble Model for Multimodal Malware Detection

Sang-Yong Choi*, Chul-Hee Lee*

*Professor, Dept. of Cloud Security, Korea Polytechnics, Daejeon, Korea

[Abstract]

As the cyber threat landscape advances, the importance of research on detecting obfuscated or packed malware variants across cross-platform environments—such as PE, ELF, Mach-O, and DEX—is increasing. However, existing techniques have limitations of false positives and false negatives due to dependency on a specific OS and overfitting of single models. To overcome this, this study proposes a novel malware detection model that combines multimodal feature fusion and a hybrid ensemble. The proposed model maps Opcode/API sequences extracted via format-independent static parsing and local Shannon entropy into a 256×256 3-channel heatmap image. Subsequently, by constructing a total of 528-dimensional multimodal features through the concatenation of SE-ResNet-based global token frequencies (512-dimensional) and entropy and structural statistics (16-dimensional), we maximized detection stability and performance using a high-performance GBDT Stacking Meta-Learner. As a result of 100-run Monte Carlo validation experiments, the proposed model achieved an average accuracy of 97.28%, an F1-Score of 97.18%, and a ROC-AUC of 0.9957, verifying that its performance outperforms existing single-based detection models.

▸ **Key words**: Malware, Entropy, Ensemble learning, Residual Network, Multimodal

[요 약]

사이버 위협 환경이 고도화됨에 따라 PE, ELF, Mach-O, DEX 등 교차 플랫폼 환경에서 난독화되거나 패킹된 변종 악성코드를 탐지하는 연구의 중요성이 증가하고 있다. 하지만 기존 기법은 특정 OS 종속성, 단일 모델의 과적합으로 인한 오탐 및 미탐의 한계를 지닌다. 이를 극복하기 위해 본 연구는 멀티모달 피처 융합과 하이브리드 앙상블을 결합한 새로운 악성코드 탐지 모델을 제안한다. 제안하는 모델은 포맷 독립적 정적 파싱으로 추출한 Opcode/API 시퀀스와 국소 Shannon 엔트로피를 256×256의 3채널 히트맵 이미지로 매핑한다. 이후 SE-ResNet 기반의 전역 토큰 빈도(512차원)와 엔트로피 및 구조적 통계(16차원)를 직렬 결합한 총 528차원의 멀티모달 피처를 구축, 고성능 GBDT Stacking Meta-Learner를 이용하여 탐지 안정성과 성능을 극대화하였다. 몬테카를로 100회 반복 실험 결과, 제안 모델은 평균 정확도 97.28%, F1-Score 97.18%, ROC-AUC 0.9957의 성능을 달성하여 기존 단일 기반 탐지 모델들보다 성능이 뛰어남을 검증하였다.

▸ **주제어**: 악성코드, 엔트로피, 앙상블 러닝, 잔차네트워크, 멀티모달

• First Author: Sang-Yong Choi, Corresponding Author: Chul-Hee Lee
*Sang-Yong Choi (csyong95@gmail.com), Dept. of Cloud Security, Korea Polytechnics
*Chul-Hee Lee (cjfclrchr@gmail.com), Dept. of Cloud Security, Korea Polytechnics
• Received: 2026. 08. 10, Revised: 2026. 10. 14, Accepted: 2026. 10. 19.

Copyright © 2026 The Korea Society of Computer and Information
http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945
www.kci.go.kr

I. Introduction

지능형 지속 위협(APT) 및 사이버 범죄 조직의 공격 기법이 고도화됨에 따라 변종 악성코드의 발생 빈도는 가속화되고 있다. 하지만 전통적인 시그니처 기반 탐지 체계는 다형성 및 변형 기법에 노출될 경우 쉽게 우회될 수 있으며, 동적분석은 가상환경 탐지 기술 및 높은 연산량으로 인해 대용량 바이너리 스트림 검증에 한계를 보인다 [1-2]. 이러한 한계를 극복하기 위해 바이너리 바이트 스트림을 이미지로 시각화 매핑하여 딥러닝 모델을 적용하는 연구가 주목받고 있다[3-5]. 최근 여러 연구들에서도 바이너리 이미징 및 미세조정 CNN 기법을 활용한 악성코드 탐지 모듈을 제안하였으나, 단순 바이트 매핑 방식은 패킹 및 암호화 쉘코드 영역을 효과적으로 탐지하기에는 구조적 한계가 존재한다[6]. 또한, 기존 연구들은 주로 Windows PE 포맷 분석에 치중하여 Linux(ELF), MacOS(Mach-O), Android(DEX) 등 다양한 운영체제 포맷을 통합 검증하지 못하는 한계성을 가지고 있다[7-8]. 더불어, 단일 딥러닝 신경망이나 단일 결정 트리 알고리즘에 의존할 경우, 특권 시스템 유틸리티나 정교하게 위장된 APT 바이너리와 같은 경계 샘플에 대하여 오탐과 미탐이 증가하는 문제가 발생한다[9-10].

본 연구에서는 이와 같은 한계점을 해결하기 위해 다중 포맷 바이너리 정적 파싱, 국소 Shannon 엔트로피 기반 3채널 히트맵 시각화, ResNet-SE CNN 특징 추출, 전역 수치 피처 직렬 결합, 그리고 하이브리드 가중 앙상블을 융합한 "하이브리드 앙상블 모델"을 제안한다. 제안하는 하이브리드 앙상블 모델은 크게 4가지 특징이 있다. 첫째, 교차 플랫폼 다중 포맷 통합 정적 파싱을 지원한다. PE, ELF, Mach-O, DEX 등에 대한 단일 파싱 엔진을 구현하여 포맷 종속성 없이 Opcode 및 API 시퀀스를 수집한다. 둘째, 국소 Shannon 엔트로피 3채널 기반 구조적 히트맵을 사용한다. 이는 단순 바이트 정렬을 넘어 Red(Opcode/API ID), Green (512bytes Shannon 엔트로피), Blue (토큰 속성 지표) 등 3개 채널로 연산되는 256×256 해상도의 히트맵으로 구성된다. 이때, 평가 단계의 데이터 누출을 차단하기 위해 모델의 어휘 사전은 학습 데이터 내부의 빈도수만으로 독립 구축한다. 그리고 제안하는 모델은 528차원 멀티모달 피처 융합 체계를 적용하였다. SE(Squeeze-and-Excitation) 모듈이 전역 토큰 빈도 분포 수치(512차원)와 엔트로피 및 구조적 통계 요약 정보(16차원)를 직렬 결합하여 이미지 리사이징으로 인한 정보 소실을 최소화한다. 마지막으로 독립적인 일반화 성능을 보장하기 위해 PyTorch Deep Residual MLP, Hist-GradientBoosting, ExtraTrees의 예측 확률을 결합하되, 앙상블 가중치 및 판정 임계값 튜닝은 전용 검증 세트 기반의 그리드 서치를 수행한다.

II. Preliminaries

1. Related works

1.1 Malware Detection via Visualization and Machine Learning Approaches

Nataraj et al.[1]은 바이너리 파일의 실행 코드를 8비트 무부호 정수 배열로 변환한 후, 가로 길이를 고정한 2차원 그래이스케일 이미지로 변환하여 악성코드를 자동으로 분류하는 이미지 기반 탐지 기법을 제안하였다. 연구에서는 추출된 이미지로부터 텍스처 특징량인 GIST (Generalized Search Tree) 피처 벡터를 산출하고, K-NN 분류기를 통해 악성코드 패밀리를 분류한다. 이와 같은 접근은 단순 변종 악성코드에 대해 98% 이상의 높은 분류 정확도를 기록하며 시각화 기법의 유용성을 입증하였다. 하지만, 단차원 바이트 직렬화 매핑에 의존하기 때문에 파일 크기에 따라 이미지 공간 레이아웃이 유동적으로 왜곡되는 현상이 발생할 수 있으며, 바이너리의 Opcode와 API의 맥락을 효과적으로 반영하지 못하고, 패킹 및 암호화 쉘코드가 포함된 영역을 구별해내는데 한계가 있어 결론적으로 변종 탐지에 한계를 나타낸다.

Cui et al.[3]은 기존 그래이스케일 매핑의 단조로운 표현력을 개선하기 위해 바이너리 바이트 시퀀스를 3개 그룹으로 분할하여 RGB 컬러 이미지로 매핑한 후 딥러닝 모델을 적용하였다. 컬러로 변환된 악성코드 이미지 데이터셋에 대해 CNN기반의 특징 추출 알고리즘을 적용하여 변종 악성코드의 미세한 변화를 포착하는 시도를 하였다. 이는 결과적으로 분류 정확도를 대폭 향상키는 결과를 가지고 왔다. 하지만 바이너리 바이트 값을 순차적으로 3등분하여 RGB 채널에 임의 매핑하는 방식을 사용하여, 각 채널이 도메인 특화 정보(예: 엔트로피, API 토큰 ID 등)를 효과적으로 수용하지 못하고, 단순 색상 채널 할당에 불과하여 의미론적 코드 구조를 필터링하지 못하며, 이미지 크기 조정시 토큰 정보가 손실될 수 있는 한계를 가지고 있다.

Kalash et al.[4]은 바이너리 그래이스케일 이미지 데이터셋에 대하여 VGG-16 및 ResNet-50과 같은 전이학습기반의 고성능 전용 CNN 아키텍처를 적용하여 피처 추출 성능을 향상시켰다. 방대한 이미지 분류용 사전 학습 가중

www.kci.go.kr

치를 재활용함으로써 소량의 악성코드 데이터셋에서도 높은 분류 성능과 강건함을 확보할 수 있음을 보여주었다. 하지만 이 연구는 일반 이미지 분류용 대형 모델을 그대로 이식함에 따라 모델 파라미터 수가 과도하여 실시간 관제 환경에서 컴퓨팅 오버헤드가 크며, 정적 바이트 구조에만 의존하므로 정밀한 API 호출 구조 및 국소 엔트로피 변화를 측정하지 못한다.

Vasan et al.[6]은 바이너리 데이터의 이미징 시각화 처리 후 미세조정된 CNN 아키텍처 및 다양한 머신러닝 알고리즘을 구동하여 판별 성능을 다각도로 비교 분석하였다. 이미지 변환 해상도와 알고리즘 특성에 따른 분류 정확도 변동성을 정량화하고, 이미지 기반 정적 분석이 갖는 수집 및 탐지 속도상의 이점을 최대화 하였다. 하지만, 이 연구는 단일 매핑 처리된 단색, 기본 컬러 이미지 모델을 사용함에 따라 바이너리 내부의 난독화 영역 및 엔트로피 급변 구역을 독립 채널로 결합해내지 못하는 한계가 있다. 또한, PE 포맷 중심의 단일 환경 데이터셋에 치중하여 교차 플랫폼(ELF, Mach-O, DEX 등)에 대한 일반화 탐지에는 한계가 있다.

Shafiq et al.[11]은 바이너리 파일의 정적 헤더 구조, PE 섹션 정보, 임포트 API 목록 등을 정교하게 정제하여 수치화된 피처 벡터로 변환하는 정적 파이프라인을 구축하였다. 수집된 다차원 정적 피처셋에 데이터 마이닝 알고리즘을 결합하여 가볍고 빠르게 악성코드를 판별하는 정적 분석 방법론을 제안하였다. 하지만 이 연구에서는 피처 추출 과정이 PE 포맷 헤더 분석에 종속되어, ELF 등 타 포맷 처리 시 파서를 다시 구현해야 하는 한계점이 있다. 아울러 바이트의 연속적 2차원 공간 배치나 국소 엔트로피의 연속성을 반영하지 못하여 바이너리 텍스처 패턴 포착 능력이 제한적이다.

Islam et al.[12]은 단일 모델이 갖는 오탐 및 과적합 문제를 해결하기 위해 다양한 정적 및 동적 피처와 여러 머신러닝 분류기(Random Forest 등)의 예측 결과를 결합하는 복합 앙상블 학습 모델을 제안하였다. 다형성 변종 악성코드가 나타내는 데이터 변동성에 대해 단일 알고리즘 대비 안정적이고 강건한 탐지 성능을 나타냄을 증명하였다. 하지만 앙상블 모델에 입력되는 피처셋이 1차원 수치 데이터로 한정되어 있으며, 딥러닝 신경망과 트리 모델 간의 도메인 교차 가중 결합 방식이 없어 결정 경계면 샘플에 대한 판단에 한계가 있다.

Saxe & Berlin[7]은 바이너리의 PE 헤더 정보, 섹션 정보, 임포트 함수 목록, 바이트 엔트로피 히스토그램 수치를 결합한 1,024차원의 수치 벡터를 구성하고, 이를 DNN에 입력하여 정적 분석을 수행하였다. 이러한 접근으로 대규모 바이너리 데이터셋 환경에서 빠른 추론 속도와 높은 정확도를 동시에 달성하였다. 하지만 피처 집계 시 엔트로피 히스토그램과 API 출현 빈도만을 수치화하므로, 코드 내부의 제어 흐름 순서나 토큰 간의 공간적 맥락(Context)을 반영하는 데 한계가 있다. 더불어 Windows PE 파일 규격에 맞춰 파서가 고정되어 있어 타 플랫폼 바이너리 적용에 한계가 있다.

Ronen et al.[13]은 대규모 Microsoft Malware Classification Challenge 데이터셋을 바탕으로 바이너리에서 추출한 Opcode, API, n-gram 피처를 구축하고, 트리 앙상블(XGBoost 등) 알고리즘을 적용하여 정밀한 패밀리 분류를 가능하게 하고, 수치 기반 트리 모델의 우수한 판별 경계 형성 능력을 검증하였다. 하지만, 이 연구는 고차원 n-gram 피처 생성 과정에서 막대한 연산 메모리와 시간이 소요되고, 시각화 기반의 2차원 공간 어텐션 특성을 활용하지 못하여 정교하게 난독화되거나 스텔스화된 코드 구역을 직관적으로 탐지하는 데 한계를 보인다.

Venkatraman et al.[10]은 단일 모달리티의 한계를 극복하기 위해 이미지 기반 바이너리 시각화 및 하이브리드 딥러닝 구조를 활용하여 악성코드에 대한 특징 추출 성능을 크게 향상시켰다. 하지만 CNN 단일 신경망의 특징 추출에만 주로 의존하여 최종 추론을 수행하므로, 신경망 특유의 경계 샘플 과신 문제를 완전히 해결하지 못했다. 수치 기반 전역 빈도 특성을 보조 피처로 면밀하게 연결하지 않아 단일 시각화 매핑에 분류 성능이 제한된다.

Gibert et al.[14]은 방대한 최신 연구 동향 분석을 통해 신경망이 추출한 피처를 트리 기반 앙상블 분류기의 입력으로 전달하는 하이브리드 탐지 구조의 가능성을 보여주었다. 신경망과 트리 모델의 장점을 융합하면 정밀도를 극대화할 수 있지만, 단일 신경망의 공간 피처만을 제한적으로 활용하는데 그쳐, 전역 토큰 빈도 및 엔트로피 통계 수치를 수평적으로 직접 연결하는 멀티모달 피처 통합이 완벽히 이루어지지 않고 있다고 언급하고 있다.

1.2 Comprehensive Limitations of Prior Works

기존 연구들은 머신러닝으로 악성코드와 정상 바이너리를 분류하기 위해 다양한 시도를 하고 있지만, 최근의 ICT 환경 변화 등에 따라 다음과 같은 기술적 한계점이 존재한다.

가) 단일 모달리티 매핑의 정보 소실: 단순 그레이스케일 또는 RGB 매핑은 패킹, 암호화 구역이 무작위 패턴으로 표현되어 바이너리의 Opcode 및 API 정

www.kci.go.kr

보와 국소 난독화 특성을 복합적으로 표현하는 데 한계가 있다[15].

나) 특정 OS 및 포맷 종속성: 연구가 Windows PE 포맷 분석에 한정되어 Linux(ELF), macOS(Mach-O), Android(DEX) 등 다중 포맷을 지원하는 데 한계가 있다[8].

다) 단일 모델 수렴 한계 및 오류 제어 미흡: 단일 신경망 모델은 경계 샘플에서 과적합을 범하기 쉽고, 단일 트리 모델은 복잡한 패턴을 포착하기 어려워 오탐과 미탐을 상쇄하기 힘들다 [9-10].

III. System Architecture

본 연구에서 제안하는 시스템은 기존 연구의 한계점을 극복하기 위해 다중 포맷 파싱 및 엔트로피 연산, 데이터 누출 방지 멀티모달 피처 생성, ResNet-SE 기반 공간 피처 추출, 528차원 피처 직렬 결합, 그리고 하이브리드 앙상블 분류 레이어 등 4단계로 설계하였다. 전체 아키텍처는 Fig. 1과 같다.

1. Multi-Format Binary Static Parsing and Sequence/Entropy Extraction

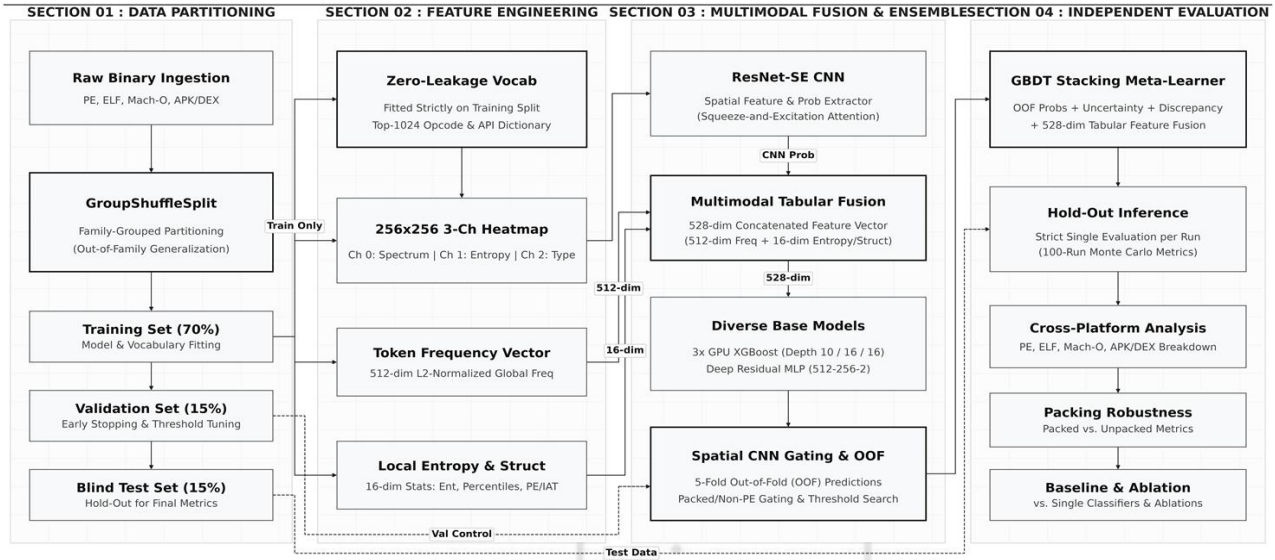
첫 번째 단계는 다양한 플랫폼 바이너리에 대한 정적 파싱 및 국소 특성 추출 과정이다. Windows PE, Linux ELF, macOS Mach-O, Android DEX 파일의 시그니처를 자동으로 식별하고, 각 파일 형식에 최적화된 독립 파서를 수행하여 Opcode 시퀀스와 API 임포트 구문을 추출한다. 추출된 시퀀스 길이는 256×256 매핑 규칙에 따라 65,536개로 절단 또는 제로 패딩 처리한다. 동시에, 처리속도 보장을 위해 메모리상에서 원본 바이너리 데이터를 512바이트 크기의 블록 슬라이딩 윈도우로 분할한 뒤, 섀넌 엔트로피 수치를 계산한다. 본 연구에서 원본 바이너리를 512바이트 크기의 블록 k로 분할하였을 때, 해당 블록 내에서의 국소 섀넌 엔트로피 H_k는 다음과 같이 정의된다.

$$H_k = -\sum_{i=0}^{255} p(i)\log_2 p(i)$$

$P(i)$: 해당 512바이트 블록 내에서 바이트 값 $i \in [0,255]$가 출현할 확률, 산출된 엔트로피 값의 범위: $0.0 \le H_k \le 8.0$ (bits)

2. Feature Engineering and 3-Channel Structural Heatmap Mapping

두 번째 단계는 추출된 정적 특성을 시각화 패턴 및 수치형 통계 공간으로 재구성한다. 데이터 누출을 예방하기 위해 학습 세트에서 추출된 토큰 분포만을 사용하여 상위 1,024개의 Opcode와 1,024개의 API로 구성된 2,048차원의 통합 어휘 사전을 구축한 후, 이를 바탕으로 256 × 256 해상도의 3채널 구조 히트맵 이미지를 생성한다. Red 채널에는 토큰 스펙트럼 수치를, Green 채널에는 256 × 256 크기로 보간 및 정규화된 국소 엔트로피 수치를, Blue 채널에는 해당 좌표의 특성 유형(Opcode 지표 128, API



www.kci.go.kr

Fig. 1. System Architecture

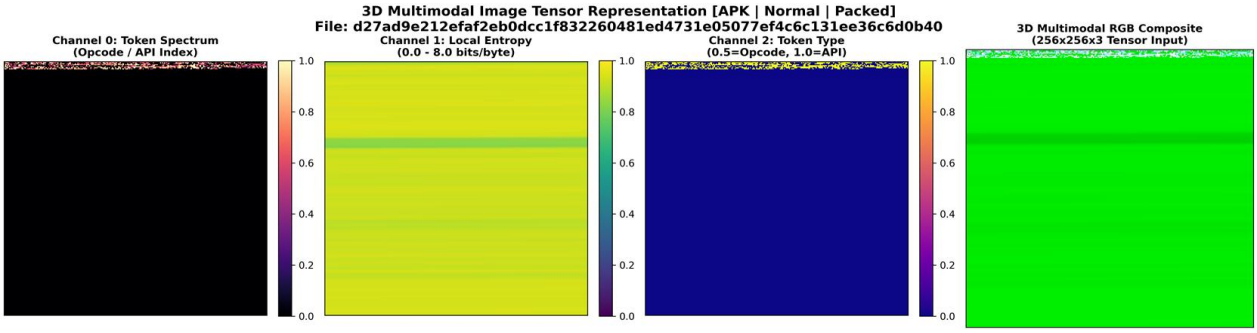


Fig. 2. Proposed 3-Channel Multimodal Heatmap Visualization: APK Normal/Packed Sample

지표 255)을 할당하여 공간 표현을 완성한다.

이후, 이미지 매핑과 독립적으로 상위 512개 토큰의 전역 출현 비율 분포 벡터(512차원)와 국소 엔트로피 요약 통계 및 구조적 특성(평균, 최대, 표준편차, 고엔트로피 비율, 백분위수, 파일 크기 및 헤더 속성 등) 16개 차원을 추가 수집하여 글로벌 특성 공간을 별도 구성한다. 3채널 멀티모달 히트맵 시각화 샘플은 Fig. 2와 같다(EX. APK)

3. Deep Spatial Feature Extraction via ResNet-SE CNN

세 번째 단계는 생성된 256×256 3채널 히트맵 이미지로부터 공간적 특징을 추출하기 위해 Squeeze-and-Excitation(SE) 모듈이 결합된 ResNet 구조 기반의 CNN 아키텍처를 학습시킨다. 본 단계의 CNN 모델은 최종 분류만을 목적으로 하지 않고, 바이너리 이미지 내의 국소 패턴과 난독화 구역을 128차원의 공간 잠재 피처로 압축 추출하는 특징 추출기로 선행 학습된다. CNN 학습 절차 구체적인 설정은 다음과 같다.

모델의 입력은 256×256×3 크기의 히트맵 텐서이며, 타겟 라벨은 정상(0)과 악성(1)이다. 학습 시 과적합을 방지하고 데이터 다양성을 확보하기 위해 35% 확률로 무작위 패치 마스킹을 수행하는 Random Cutout 데이터 증강을 적용한다. 손실 함수로는 클래스 불균형을 해소하기 위한 클래스 가중치와 함께 모델의 과신방지를 위한 레이블 스무딩(0.05)이 적용된 가중 크로스 엔트로피 손실을 사용한다.

모델 최적화는 AdamW Optimizer를 통해 수행되며, 초기 학습률(Learning Rate)은 η=0.0005, 가중치 감소 정규화 파라미터는 $\lambda = 10^{-4}$로 설정한다. 또한, 학습 안정성을 높이기 위해 매 에포크마다 코사인 어닐링 학습률 스케줄러가 연동되며 그래디언트 폭발을 방지하기 위해 Max Norm 5.0 기준의 그래디언트 클리핑을 적용한다. 학습은 최대 100에포크 동안 진행되며, 과적합을 방지하기 위해 검증 세트 기준의 손실이 15에포크 동안 개선되지 않을 경우 조기 종료가 발동되고 최적의 가중치 상태가 복원된다. 최종적으로 학습이 완료된 CNN 모델로부터 종단 분류 레이어를 거치기 직전의 활성값을 추출한 뒤, L2 정규화($p = 2$, $\dim = 1$)를 거쳐 압축된 128차원의 잠재 피처를 확보한다. 최종 앙상블 융합 단계에서는 이 공간 피처의 직접적인 수치 결합 대신, 528차원의 멀티모달 수치 피처(전역 토큰 빈도 512차원 및 엔트로피/구조적 통계 16차원)와 함께 각 베이스 분류기의 예측 확률 및 불확실성 지표가 통합되며, 패킹 여부 및 포맷 특성에 따른 Spatial CNN Gating 메커니즘을 거쳐 최종 메타 러너의 입력 피처로 전달된다.

4. Multimodal Feature Space Fusion & Hybrid Ensemble

네 번째 단계에서는 서로 다른 모달리티에서 기인한 피처 벡터들을 통합한다. 구체적으로는 전역 토큰 빈도 특징(512차원)과 엔트로피 및 구조적 통계 특징(16차원)을 직렬 결합하여 총 528차원의 멀티모달 수치 특징 공간을 구축하며, Squeeze-and-Excitation(SE) 모듈이 결합된 ResNet 기반 딥러닝 모델은 고유의 시각적 공간 잠재 피처(128차원) 추출과 더불어 Spatial CNN Gating 메커니즘을 통해 패킹 여부 및 포맷 특성에 따른 예측 확률을 산출하는 데 활용된다.

최종 분류 단계에서는 528차원의 멀티모달 피처와 베이스 모델들의 예측 확률, 불확실성, 모델 간 예측 불일치 통계량을 메타 피처로 통합하여 입력받는 고성능 GBDT Stacking Meta-Learner를 수행한다. 심층 잔차 신경망과 서로 다른 하이퍼 파라미터 설정을 가진 3개의 GPU 가속 XGBoost 모델을 베이스 분류기로 결합한 뒤, 5-Fold 교차 검증 기법을 통해 과적합을 방지하고 메타 러너를 학습시킨다. 특히, 모델의 최종 결정 경계 최적화는 오직 별도로 분리된 검증 세트상의 탐색을 통해서만 수행되어 평가 세트에 대한 모델의 일반화 성능을 담보한다.

www.kci.go.kr

IV. Experiments and Result Analysis

1. Dataset Construction and Experimental Setup

1.1 Dataset Construction

제안하는 아키텍처의 교차 플랫폼 탐지 성능 및 난독화·패킹 악성코드에 대한 강건성을 검증하기 위해 다중 포맷 데이터셋을 Table. 1과 같이 구축하였다.

- 정상(Benign) 파일: Sysinternals Live, MSYS2 MinGW-w64(Windows PE), Ubuntu Archive(Linux ELF), Homebrew(macOS Mach-O), F-Droid (Android APK/DEX) 등 검증된 공식 저장소 및 패키지 관리자에서 정상 소프트웨어 바이너리 4,235개.
- 악성(Malware) 파일: MalwareBazaar 플랫폼의 API를 연동하여 트로이목마, 랜섬웨어 등 포맷별 최신 악성코드 및 패커가 적용된 샘플 3,943개

Table 1. Dataset Construction

OS/ Platform	Malware (Unpacked/Packed)	Benign (Unpacked/Packed)
Windows PE	539 / 1,553	1,720 / 412
Linux ELF	680 / 118	547 / 454
MacOS Mach-O	55 / 246	191 / 122
Android APK	99 / 653	5 / 784
Total	3,943 (1,373 / 2,570)	4,235 (2,463 / 1,772)

1.1 Experimental Setup

모델 학습 및 평가 시, 단일 무작위 표본 분할에서 발생할 수 있는 '동일 악성코드 패밀리 변종의 학습-테스트 동시 포함(데이터 누수)' 예방을 위해 악성코드 패밀리 그룹 단위 분할을 적용하여 전체 데이터를 학습(70%), 검증(15%), 블라인드 테스트(15%)로 분할하였다.

더불어, 성능의 변동성 및 우연성을 배제하고 평가의 신뢰도를 확보하기 위해 B200 GPU 환경에서 100회의 몬테카를로 반복 실험을 수행하여 지표의 평균과 표준편차를 도출하였다. 실험에 사용된 하드웨어 및 소프트웨어 사양은 Table 2.와 같다.

Table 2. Summary of Experimental Setup

Category	Parameter / Component	Specification / Value
Hardware	GPU	NVIDIA B200 GPU
Software	Operating System	Linux (Ubuntu Server)
	Frameworks	PyTorch, Scikit-learn, PIL, Matplotlib

2. Result Analysis

[Result Analysis 1] 100회 반복 실험 통계적 종합 성능

바이너리 패밀리 그룹 분할과 데이터 누수 방지 기반으로 몬테카를로 100회 반복 실험 결과는 Fig. 3과 같으며, 제안 모델의 블라인드 테스트 세트 평균 및 표준편차는 Table. 3과 같다.

결과적으로 제안한 모델은 동일 변종의 도움 없이 순수하게 미지의 패밀리를 분류해야 하는 테스트 환경에서도 평균 97.28%의 뛰어난 정확도와 매우 안정적인 변동성(±0.40%)을 입증하여 실제 현업 적용 가능한 일반화 성능을 달성하였다.

Table 3. 100-Run Statistical Performance Summary

Evaluation Metric	Evaluation Result
Accuracy	97.28% ± 0.40%
Precision	97.28% ± 0.67%
Recall	97.08% ± 0.77%
Specificity	97.46% ± 0.64%
F1-Score	97.18% ± 0.42%
ROC_AUC	0.9957 ± 0.0014

[Result Analysis 2] 비교 모델 및 제거 실험

본 제안 모델 아키텍처의 기여도를 검증하기 위해 동일한 100회 반복 분할 조건하에 기존 개별 분류기 및 구성 요소 제거 실험을 수행하였다. 실험 결과, Table. 4와 같이 제안하는 하이브리드 모델은 어떠한 단일 분류 모델보다 우수한 성능을 보였다. 특히 모달리티 제거 실험에서 시각 공간 피처, 토큰 빈도 벡터, 엔트로피 통계 중 단 하나라도 누락될 경우 전체 정확도가 유의미하게 하락(94.89% ~ 97.03%)하여 제안하는 모든 피처를 결합한 멀티모달 융합의 성능을 입증하였다.

Table 4. Baseline Comparison & Ablation Study

Method	Evaluation Result
Baseline - CNN Only	Acc 95.22% (± 0.62%), F1 95.04%
Baseline - MLP Only	Acc 95.59% (± 0.62%), F1 95.45%
Baseline - HistGB Only	Acc 96.97% (± 0.45%), F1 96.87%
Baseline - ExtraTrees Only	Acc 96.94% (± 0.43%), F1 96.83%
Baseline - Random Forest	Acc 96.90% (± 0.46%), F1 96.79%
Baseline - Logistic Regression	Acc 91.33% (± 0.78%), F1 90.97%
Ablation Spatial CNN (Only Freq + Entropy)	Acc 97.03% (± 0.41%), F1 96.92%
Ablation Token Freq (Only CNN + Entropy)	Acc 94.89% (± 0.69%), F1 94.72%
Ablation Entropy (Only CNN + Freq)	Acc 94.90% (± 0.68%), F1 94.72%
Full Proposed Mode	Acc 97.28%, F1 97.18%

www.kci.go.kr

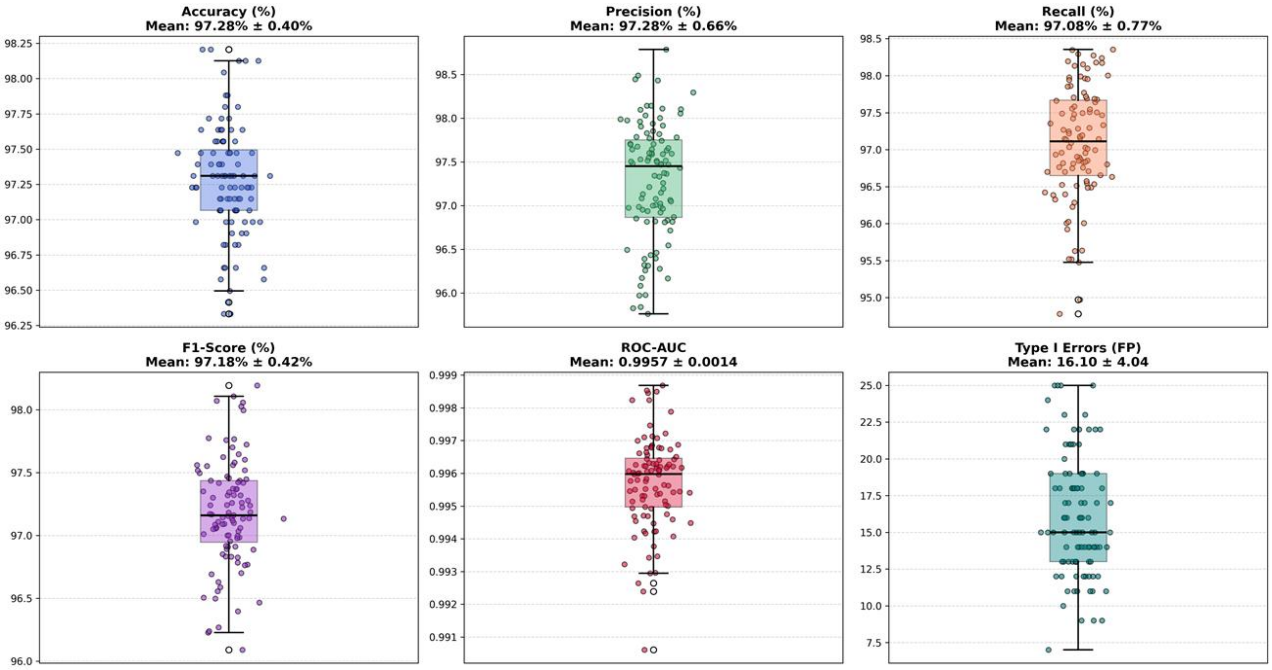


Fig. 3. Evaluation 100 runs Distribution

[Result Analysis 3] 교차 플랫폼 및 난독화/패킹 강건성

논문의 주요 기여인 이기종 포맷 지원과 난독화 강건성을 확인하기 위해 파일 포맷별 및 패킹 여부별 성능을 분석한 결과 fig. 4와 같이 PE 포맷뿐만 아니라 ELF 및 APK 포맷에서도 안정적인 탐지율을 확인하였다. 특히 국소 엔트로피 특성 채널을 시각화에 융합한 결과, 코드가 암호화되거나 압축된 표본 집단에서도 96.41%의 높은 탐지성능을 유지하였다.

```
================================================================
       CROSS-PLATFORM PERFORMANCE BREAKDOWN BY EXECUTABLE FORMAT
================================================================
  Format Avg Test Samples    Accuracy (%)   F1-Score (%)
     APK              225  95.69% ± 1.29% 95.40% ± 1.40%
     ELF              268  98.63% ± 0.69% 98.45% ± 0.79%
Fallback                5 92.60% ± 11.17% 95.77% ± 6.66%
  Mach-O               93  91.37% ± 2.72% 91.02% ± 3.11%
      PE              633  98.18% ± 0.48% 98.16% ± 0.49%
================================================================

================================================================
       PACKING & OBFUSCATION ROBUSTNESS BREAKDOWN
================================================================
Packing Category Avg Test Samples   Accuracy (%)   F1-Score (%)
         Packed              651 96.41% ± 0.61% 96.96% ± 0.55%
       Unpacked              575 98.26% ± 0.51% 97.58% ± 0.70%
================================================================
```

Fig. 4. Per-Format & Per-Packing Breakdown

[Result Analysis 4] 컴퓨팅 자원 효율성 및 오분류 분석

실시간 보안 관제 시스템에 적용 가능함을 입증하기 위해 모델의 연산 효율성 및 병목을 측정하였다. 측정 결과 Fig. 5와 같이 제안한 모델은 높은 처리량을 보여 대규모 분석에 적합함을 확인하였다. 다만, Preprocessing Time per Binary File (1.01 ms/file)의 경우 디스크의 개별 바이너리를 매번 파싱하지 않고, 사전 추출 캐시를 통해 메

```
================================================================
      COMPUTATIONAL EFFICIENCY & RESOURCE PROFILING TABLE
================================================================
- Preprocessing Time per Binary File   : 1.01 ms/file
- CNN Training Time per Iteration      : 182.2 seconds
- Test Inference Latency per Sample    : 0.111 ms/sample
- System Throughput (Inference FPS)    : 9046.8 samples/sec
- ResNet-SE CNN Parameter Count        : 7,055,778 parameters (26.92 MB)
- Residual MLP Parameter Count         : 539,650 parameters (2.06 MB)
- Peak RAM Consumption                 : CPU RAM: 5856.1 MB | GPU VRAM: 23803.8 MB
================================================================
```

Fig. 5. COMPUTATIONAL EFFICIENCY & RESOURCE PROFILING

```
[*] Parsing progress: 8050/8178 files parsed (98.4%) | 6.7 samples/s | ETA: 0m 19s
[*] Parsing progress: 8100/8178 files parsed (99.0%) | 6.7 samples/s | ETA: 0m 11s
[*] Parsing progress: 8150/8178 files parsed (99.7%) | 6.7 samples/s | ETA: 0m 04s
[*] Parsing progress: 8178/8178 files parsed (100.0%) | 6.8 samples/s | ETA: 0m 00s
[+] [💾 CPU PHASE 1/2 COMPLETE] One-time parsing complete: 8178 samples parsed in 1208.25s (147.74 ms/file).
[+] Saved high-speed compact dataset cache to 'parsed_samples_cache.pkl' for instant future executions!
```

Fig. 6. Preprocessing latency for file

모리상에서 직접 배치 텐서로 변환하여 전처리 지연이 1.011 ms로 측정되었다. 실제 최초 1회 파싱 시간은 Fig. 6과같이 총 8,178개의 바이너리 파싱에 1208.25초가 소요되어, 파일당 처리시간은 147.74ms로 실시간에 가까운 성능을 보여주고 있다. 그리고, Inference Latency (0.111 ms for sample / 9,648.8 FPS)의 경우 테스트 세트의 특징 추출 및 분류기 판별을 CUDA VRAM 상에서 일괄 병렬 행렬 연산으로 실행하여 극히 짧은 시간에 처리되고 있다. 또한, Fig. 7에서 확인되는 바와 같이 100회 평균 발생한 1종 오류는 16.1건(±1.3), 2종 오류는 17.3건(±1.4) 발생하였다. 오분류 원인을 분석한 결과, 주로 .NET 인젝션 프레임워크나 비표준 패킹이 적용된 특권 유틸리티들이 난독화된 악성 바이너리의 엔트로피 분포와 구조적 유사성을 띠어 발생한 것으로 확인되었다.

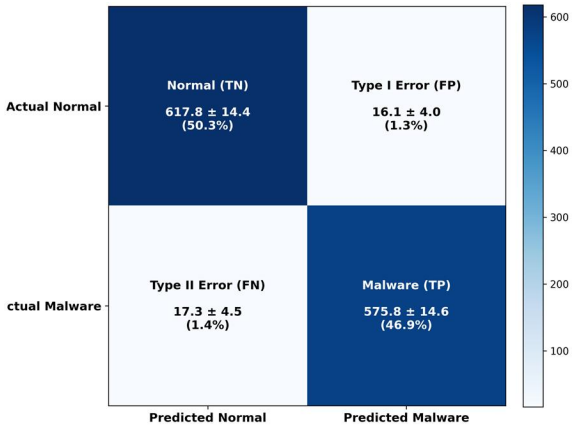


Fig. 7. Aggregated Confusion Matrix

V. Conclusions

본 연구에서는 교차 플랫폼 다중 포맷 바이너리의 정적 분석 시퀀스 및 국소 Shannon 엔트로피 특성을 3채널 구조적 히트맵으로 매핑하고, 하이브리드 앙상블을 융합한 악성코드 탐지모델을 제안하였다.

특히, 모델 학습 시 데이터 누출문제를 차단하기 위해, 어휘 사전 및 특징 파라미터를 학습 데이터에 국한하여 추출하고 검증 데이터에서 가중치를 튜닝하는 평가 체계를 도입하였다. 악성코드 패밀리 동질성을 통제한 100회의 몬테카를로 반복 평가 결과, 제안 모델은 평균 정확도 97.28%, F1-Score 97.18%, ROC-AUC 0.9957을 달성하였다. 이는 기존의 단일 기반 탐지 모델들의 성능을 크게 상회하며, 시각 및 수치 피처의 모든 요소가 결합될 때 최적의 성능을 냄을 실험을 통해 검증하였다. 나아가 난독화된 패킹 코드 영역(96.41%)과 대부분의 주요 OS 아키텍처 환경(PE 98.18%, ELF 98.63%, APK 95.69%)에서도 높은 탐지 정확도를 유지하였으며, 우수한 컴퓨팅 자원 효율성을 확인하여 실제 통합 보안 관제 체계에 접목할 수 있는 기술적 타당성을 제공하였다.

향후 본 연구를 확장하여 제안한 시스템을 실무적 보안 관제 업무에 활용할 수 있도록 고해상도 이미지 매핑을 통한 초미세 바이너리 패턴 추출, 어휘 사전 크기 확장 및 파라미터 스케일업, 대용량 바이너리의 실시간 병렬 추론 엔진을 구축하여 처리성능을 극대화할 예정이다. 또한, 공격자가 노이즈 바이트를 주입하는 등과 같은 적대적 기법에 대응하기 위해 FGSM/PGD 공격 모델을 구축하고, Adversarial Training 기법을 도입하여 모델 견고성을 체계적으로 강화할 예정이다.

www.kci.go.kr

ACKNOWLEDGEMENT

This work was supported by the National AI Project funded by the Ministry of Science and ICT (MSIT), Korea

REFERENCES

[1] L. Nataraj, V. Yegneswaran, P. Porras, and R. Zhang, "A Comparative Assessment of Malware Classification Using Binary Texture Analysis," Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence (AISEC), pp. 21-30, Chicago, IL, USA, October 2011. DOI: 10.1145/2046684.2046689

[2] A. Moser, C. Kruegel, and E. Kirda, "Limits of Static Analysis for Malware Detection," 23rd Annual Computer Security Applications Conference (ACSAC), pp. 420-430, Miami Beach, FL, USA, December 2007. DOI: 10.1109/ACSAC.2007.21

[3] Z. Cui, F. Xue, X. Cai, Y. Cao, G. G. Wang, and J. Chen, "Detection of Malicious Code Variants Based on Deep Learning," IEEE Access, Vol. 6, pp. 38856-38866, July 2018. DOI: 10.1109/tii.2018.2822680

[4] M. Kalash, M. Rochan, N. Mohammed, N. D. Bruce, Y. Wang, and F. Iqbal, "Malware Classification with Deep Convolutional Neural Networks," 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS), pp. 1-5, Paris, France, February 2018.

[5] H. Kim, S. Han, S. Lee, and J. R. Lee, "Visualization of Malwares for Classification Through Deep Learning," Journal of Internet Computing and Services, vol. 19, no. 5, pp. 67-75, Oct. 2018. DOI: 10.7472/jksii.2018.19.5.67.

[6] D. Vasan, M. Alazab, S. Wassan, H. Naeem, B. Safaei, and Q. Zheng, "IMCFN: Image-based Malware Classification Using Fine-tuned Convolutional Neural Network Architecture," Computer Networks, Vol. 171, pp. 107138, April 2020. DOI: 10.1016/j.comnet.2020.107138

[7] J. Saxe, and K. Berlin, "Deep Neural Network Based Malware Detection Using Two Dimensional Binary Program Features," 10th International Conference on Malicious and Unwanted Software (MALWARE), pp. 11-20, Fajardo, PR, USA, October 2015. DOI: 10.1109/MALWARE.2015.7413680

[8] A. Cozzi, M. Graziano, D. Fratantonio, and D. Balzarotti, "Understanding Linux Malware," IEEE Symposium on Security and Privacy (S&P), pp. 161-175, San Francisco, CA, USA, May 2018. DOI: 10.1109/SP.2018.00054

[9] E. Raff, J. Sylvester, and C. Nicholas, "Learning the PE Header, Malware Detection with Minimal Domain Knowledge," ACM Transactions on Privacy and Security, Vol. 21, No. 3, pp. 1-28,

June 2018. DOI: 10.1145/3128572.3140442

[10] S. Venkatraman, M. Alazab, and R. Vinayakumar, "A Hybrid Deep Learning Image-based Analysis for Effective Malware Detection," Journal of Information Security and Applications, Vol. 47, pp. 377-389, August 2019. DOI: 10.1016/j.jisa.2019.06.006

[11] M. Z. Shafiq, S. M. Tabish, F. Mirza, and M. Farooq, "PE-Miner: Mining Structural Information to Detect Malware Executables in Realtime," 12th International Symposium on Recent Advances in Intrusion Detection (RAID), pp. 164-183, Saint-Malo, France, September 2009. DOI: 10.1007/978-3-642-04342-0_7

[12] R. Islam, R. Tian, L. M. Batten, and S. Versteeg, "Classification of Malware Based on Integrated Static and Dynamic Features," Journal of Network and Computer Applications, Vol. 36, No. 2, pp. 646-656, March 2013. DOI: 10.1016/j.jnca.2012.10.004

[13] R. Ronen, M. Radu, C. Feuerstein, E. Yom-Tov, and Y. Mansour, "Microsoft Malware Classification Challenge," arXiv preprint arXiv:1802.10135, pp. 1-10, February 2018.

[14] D. Gibert, C. Mateu, and 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, pp. 102526, March 2020. DOI: 10.1016/j.jnca.2019.102526

[15] M. Lyda, and J. Hamrock, "Using Entropy Analysis to Find Encrypted and Packed Malware," IEEE Security & Privacy, Vol. 5, No. 2, pp. 40-45, March 2007. DOI: 10.1109/MSP.2007.48

Authors



Sang-Yong Choi received his B.S. degree in Mathematics and M.S. degree in Computer Science, both from Hannam University in 2000 and 2003, and Ph.d degree in Interdisciplinary of Information Security from Chonnam National University in 2014. Dr. Choi is a assistant professor at Dept. of Cloud Security in Korea Polytechnics, Daejeon, Korea. His research interests are in security education, web security, network security and cloud computing security.



Chul-Hee Lee received the B.S. degree in information computer science from Konyang University, Korea, in 2015, and the M.S. degree in information security from Sejong Cyber University, Seoul, Korea, in 2026. He is currently a visiting professor with the Department of Cloud Security, Korea Polytechnics, Daejeon, Korea. His research interests include data and 3D visualization, cloud security, and prompt engineering for vibe coding.

www.kci.go.kr