IVI 시스템 포렌식을 위한 파일 파편 유형 분류

이승민[1] 노경민[2] 정지헌[3] 박수현[1] 조성제[1]
단국대학교 소프트웨어학과[1], 단국대학교 사이버보안학과[2], 단국대학교 컴퓨터학과[3]
{seungmin_lee, imsie1, wlgjsjames7224, parksh, sjcho}@dankook.ac.kr

Classifying File Fragment Types for IVI System Forensics

Seungmin Lee[1] Kyoungmin Roh[2] Jeehun Jung[3] Suhyeon Park[1] Seong-je Cho[1]
Department of Software Science, Dankook University[1]
Department of Cybersecurity, Dankook University[2]
Department of Computer Science and Engineering, Dankook University[3]

요 약

차량의 IVI(In-Vehicle Infotainment) 시스템에 저장된 데이터는 사고 발생 시 운전자의 행위 분석, 운행 기록, 차량 범죄 수사 등 차량용 디지털포렌식에서 중요한 증거 자료가 될 수 있다. IVI 시스템을 대상으로 차량 포렌식에서 삭제된 파일 데이터를 복구하여 주요한 아티팩트(artifacts)를 추출하여 분석하는 파일 카빙(file carving)이 주요 이슈다. 파일 카빙 기법에서 중요한 과정은 복구 대상 파일의 파편 타입(fragment type)을 식별·분류하는 것이다. 본 논문에서는, IVI 시스템에서 사용되는 주요 파일들에 대한 파편들을 엔트로피 기반으로 1차 분류하고, 다음으로 머신러닝 기반으로 2차 분류하는 기법을 제안한다. 특징 정보로는 각 파편의 엔트로피와 바이트 스트림 기반 N-gram을, 분류모델로는 RF(random forest)와 1D-CNN(one-dimensional convolutional neural network)을 사용한다. 2가지 단계로 대상 파일들의 파편들을 분류하는 실험을 통해 제안 기법의 성능을 평가하고 한계점도 분석한다.

1. 서 론

디지털포렌식은 디지털 증거를 수집, 분석, 보존하여 법적 증거로 활용하는 과학적 절차이다[1]. 최근 자동차의 전장화 및 지능화가 가속화되면서 IVI(In-Vehicle Infotainment) 시스템은 단순한 정보 제공을 넘어 차량 운행 이벤트 및 사용자 행위에 대한 방대한 데이터를 생성하고 저장하는 핵심 장치로 부상하고 있다[2]. 이러한 IVI 시스템은 사고 발생 시 운전자의 행위 분석, 운행 기록, 차량 관련 범죄 수사 등과 같은 차량용 디지털포렌식에서 중요한 증거 자료가 될 수 있다[3]. 하지만 IVI 시스템의 한정된 저장 공간이라는 특성상 데이터가 단편화되거나 손상되기 쉬우며, 파일 시스템 정보가 온전히 보존되지 않는 경우가 많아 기존의 파일 시스템 기반 포렌식 기법으로 데이터 복구하기 어렵다는 한계가 있다.

파일 카빙(file carving)은 파일 시스템의 메타데이터에 의존하지 않고 원시(raw) 데이터에서 특정 패턴 혹은 구조적 특징을 분석하여 손상되거나 삭제된 파일을 복구하는 기법이다[4, 5]. 파일 카빙 과정에서 복구 대상 파일 파편들 기반으로, 각 파편이 어떤 파일 유형에 속하는지 정확하게 분류하는 것은 복구 성공률과 밀접하게 관련되어 있다. 특히, IVI 시스템에서 주로 생성되는 로그 파일(txt, log, dat, pcm)과 데이터 파일(sqlite, xml)들은 그 종류가 다양하기에, 관련 파일 파편들을 효율적으로 분류하는 기술은 IVI 포렌식에서 매우 도전적 과제이다.

한편, IVI 시스템의 저장 장치는 과거 HDD를 사용하는 방식에서 대부분 SSD로 전환되고 있으며, SSD의 특성상 데이터 삭제 및 할당 방식이 HDD와 달라 파편화 양상 또한 복잡해진다[6]. 예를 들어, SSD에서는 웨어 레벨링(wear leveling) 및 가비지 컬렉션(garbage collection)과 같은 기술로 인해 데이터가 예상하지 못한 방식으로 재배치되거나 삭제될 수 있다.

이에 본 연구는 IVI 시스템 환경에서 데이터 파편만을 활용하여 파일 유형을 효율적으로 분류하는 머신러닝 기반 기법을 제안한다. 먼저, IVI 시스템에서 주로 사용되는 파일 유형의 파편을 수집하여 특징을 추출한다. 이후 특성과 부합한 적합한 모델을 설계하여 파일 분류모델의 정확도와 효율성을 향상한다. 이를 통해 IVI 시스템에 특화된 파일 카빙 연구의 기초를 마련하고자 한다.

2. 관련 연구

파일 카빙 및 파일 파편 유형을 분류하는 방식에 대해, 다양한 연구들이 수행됐다. 초기 연구들은 주로 파일의 헤더(header) 및 푸터(footer) 시그니처(signature)를 기반으로 파일 복구하고 유형을 식별하는 방식에 집중했다[7]. 이는 간단하고 효율적이지만, 시그니처가 손상되거나 존재하지 않는 파편들에 대해서는 어려움이 있다. 또한 고유 시그니처 정보가 없거나, 또는 다른 파일들의 시그니처와 중첩될 때는 분류 정확성이 떨어진다.

이에 최근에는 머신러닝 기법을 활용하여 파일 유형을

분류하는 연구가 진행되고 있다. 대표적으로 파일의 바이트 시퀀스, 엔트로피(Entropy), N-gram 등과 같은 여러 특징 정보를 추출하여 SVM(Support Vector Machine), Naive Bayes, DT(Decision Tree) 등의 모델을 활용하는 연구들이 수행되었다[8-10].

CNN(Convolutional Neural Network)을 활용한 파일 유형 분류 연구도 수행되었다. 특히 1D-CNN은 시퀀스 데이터 분석에 강점을 가지며, 파일의 바이너리 데이터를 1차원 시퀀스로 처리하여 특징을 분석 및 분류하는데 효과적이다[11]. 이는 기존의 수동적 특징 정보 추출 방식의 한계를 극복하고, 파일의 파편과 같은 복잡하고 다양한 패턴을 가진 데이터에서도 높은 분류 성능을 기대할 수 있다.

김다은 등[12]은 안드로이드 기반 IVI 시스템에서 머신러닝을 활용한 파일 파편 유형 분류를 수행하기 위한 관련 연구를 조사하고 방법론을 제시했다. 또한 4개의 차량 IVI 시스템에서 확인할 수 있는 파일 유형을 종합하여 총 114개의 파일 유형이 존재하고 있음을 확인했다. 또한 4개의 IVI 시스템들은 안드로이드 Jellybean 버전과 Kitkat 버전 기반이며 사용하는 파일 시스템은 ext4임을 밝혔다.

IVI 시스템 환경에서 파일 카빙과 같은 포렌식 연구는 아직 초기 단계에 있다. IVI 시스템은 고유의 운영체제, 파일 시스템의 사용으로 인해 일반적인 PC 환경에서 연구된 파일 파편 분류 기법을 적용하는 것에 한계가 있다.

3. 데이터셋 및 파편화

이 장에서는 머신러닝 학습을 위한 데이터 구성 및 전처리 과정에 대해 설명한다. 본 논문에서는 govdocs1 데이터셋[13]을 사용한다. govdocs1은 2009년에 제작되었으며 “.gov” 도메인의 웹 서버에서 Yahoo와 Google 검색 엔진을 활용하여 지정된 파일 형식을 가진 파일들을 포함한다. 이 데이터셋은 100만 개의 파일을 포함하는데, 상세하게는 1,000개의 디렉터리마다 1,000개의 파일을 보유하고 있다. 또한 govdocs1은 총 213개 종의 다양한 파일 유형을 보유하고 있지만 파일 유형 간 불균형한 분포로 인해 활용 단계에서는 각 유형이 충분한 샘플을 보유하고 있는지 확인이 필요하다.

3.1 데이터 전처리

본 연구는 안드로이드 기반 IVI 시스템에서 사용되고 있는 파일 유형에 대해 분류하기 위해, govdocs1이 보유하고 있는 100만 개의 파일들의 유형과 대상 IVI 시스템에서 확인된 114개의 파일 유형의 교집합을 분석하였다. 수행 결과 13개의 공통된 파일 유형을 탐색하였으며, 그 결과는 표 1과 같다.

govdocs1 데이터셋은 파일 유형별 개수 분포가 불균형하다. 그리고 머신러닝을 통한 파일 유형 분류를 위해서는 유형마다 충분한 데이터 개수가 요구된다. 이에 데이터 개수가 충분하지 않은 클래스인 .java, .tmp, .ttf, .zip, .bin은 본 연구에서 제외한다. 최종 선정된 8개의 파일 유형(.html, .jpg, .txt, .gif, .xml, .gz, .log, .swf)에 대해 분류하는 연구를 수행한다.

표 1. IVI 시스템과 govdocs1 데이터셋의 공통 파일 유형

Type	Count
.html	207,857
.jpg	109,164
.txt	74,022
.gif	36,132
.xml	32,929
.gz	13,703
.log	9,643
.swf	3,458
.java	288
.tmp	137
.zip	10
.ttf	7
.bin	1

본 논문은 IVI 시스템용 파일 파편 분류를 위한 초기 연구 단계이므로, HDD 환경을 기반으로 데이터셋을 구성했다. HDD는 섹터(sector) 단위로 데이터를 저장하며, 파일 할당 단위는 클러스터(cluster)이다. 파일 삭제 시에는 해당 클러스터의 할당 정보만 해제되고 실제 데이터는 남아있게 되므로, 파일 파편은 클러스터 단위로 존재할 수 있다. 하지만 현대 대부분의 IVI 시스템은 SSD로 전환되고 있다. SSD는 페이지(page) 단위로 데이터를 읽고 쓰며, 블록(block) 단위로 데이터를 삭제한다. 또한 웨어 레벨링, 가비지 컬렉션과 같은 기술들로 인해 데이터가 물리적으로 재배치되거나 삭제될 수 있기에, HDD와는 다른 파편화 형태를 가질 수 있다. 그렇기에 SSD에서는 페이지 혹은 블록 단위로 파편화가 발생할 수 있다.

3.2 데이터 파편화

HDD 환경을 가정하여 파일 파편만을 가지고 파일 유형을 예측할 수 있는지를 확인하기 위해 분리한 데이터에 대해 파일 파편을 생성했다. 이때 파일 헤더와 푸터 정보는 파일 유형을 식별하는 중요한 정보이지만, 본 논문에서는 위 정보들을 제외한 영역의 파편들만을 대상으로 하여 파일 유형을 분류할 수 있는지를 평가한다. 즉, 실험 대상 8개 유형을 가진 파일들에 대해 헤더와 푸터를 제거한 다음 1,024바이트 크기의 파편들을 생성했다.

4. 특징 추출 및 사용 모델

4.1 특징 추출

머신러닝을 적용하여 파일 파편만으로 파일 유형을 분류하기 위해서, 적합한 특징 추출이 분류모델의 성능에 큰 영향을 끼친다. 이에 본 논문에서는 파일 파편의 바이트 스트림으로부터 다음과 같은 특징 정보들을 추출하

여 사용한다.

a) 엔트로피(Entropy): 파일 파편의 정보량을 나타내는 정보로 데이터의 무작위성 혹은 암호화/압축 정도를 측정한다. 엔트로피 값이 클수록 데이터의 무작위성과 암호화/압축이 되었을 가능성이 높다. 본 논문에서는 각 파일 파편의 엔트로피 값을 계산한 다음 [14]에 기반한 7.0을 기준으로 High Entropy와 Low Entropy 파편 그룹으로 분류한다.

b) N-gram: 파일 파편 내에서 연속적으로 나타내는 N개의 바이트 스트림의 출현 빈도를 분석하는 기법이다. 이는 파일 유형별로 특정 바이트 시퀀스(sequence)가 자주 나타나는지를 분석할 수 있다. 예를 들어 txt 파일에서는 "0D 0A"와 같은 줄 바꿈 문자가 자주 나타나는 특징이 있는 것처럼 각 파일에서 자주 나타나는 바이트 패턴이 존재한다. N-gram은 이러한 짧은 바이트 패턴들을 '한 단위'로 인식하여 파일 파편 특징으로 사용한다. 본 논문에서는 3~5바이트 길이를 추출하여 사용한다.

4.2 분류 기법 및 모델

파일 파편의 유형을 분류하기 위해 TF-IDF(Term Frequency-Inverse Document Frequency), RF(Random Forest) 그리고 1D-CNN(One-Dimensional Convolutional Neural Network) 모델을 사용한다.

4.2.1 TF-IDF

TF-IDF는 텍스트 마이닝에서 문서 내 단어 중요도를 측정하는 데 주로 사용되는 통계적 가중치 기법이다. 본 논문에서는 파편의 바이너리 데이터를 텍스트 문서로 처리하여 TF-IDF를 사용한다. 이때 파일 파편 하나는 "문서"가 되고 N-gram (3~5바이트)는 "단어"가 된다. TF-IDF는 다음과 같은 요소로 구성된다.

- TF(Term Frequency): 특정 N-gram이 파편 내에서 얼마나 자주 나타나는지를 측정한다. 예를 들어 0D 0A'이라는 N-gram이 많이 측정된다면, 해당 N-gram의 TF는 증가한다.
- IDF(Inverse Document Frequency): 특정 N-gram이 본 파일을 제외한 나머지 파일에서 몇 번 사용되었는지를 측정한다. 만약 '0D 0A'가 txt 파일 파편에서는 자주 관측되지만 다른 jpg, html에서는 거의 나타나지 않는다면, IDF 값을 상승한다. 따라서 이 N-gram은 txt 파일 유형을 구분하는 데 중요한 특징인 것을 의미한다.

4.2.2 Random Forest (RF)

RF는 여러 개의 결정 트리를 구성하여 분류 또는 회귀 분석에 사용하는 앙상블 학습 방법이다. 각 결정 트리는 무작위로 선택된 샘플과 특징을 사용하여 학습하며 최종 예측은 개별 트리의 예측을 분류 혹은 회귀하여 결정한다. 따라서, RF는 과적합에 강인하고 특징 중요도(Feature Importance)를 파악하는 데 장점이 있다.

이에 Low Entropy 파편들을 TF-IDF를 특징으로 하여 RF로 학습한다면 특정 바이트에 높은 가중치를 부여할 수 있고 RF는 이를 비선형적으로 조합하여 효과적으로 분류할 수 있다.

4.2.3 1D-CNN

1D-CNN은 시퀀스 데이터 분석에 강점을 가지며, 파일의 바이너리 데이터를 1차원 시퀀스로 간주하여 특징을 자동으로 추출하고 분류한다. 1D-CNN은 다음과 같은 구성요소를 가진다.

- 입력층(Input Layer): 파일 파편의 바이트 시퀀스를 입력으로 받는다.
- 합성곱층(Convolutional Layers): 여러 필터를 사용하여 입력 데이터로부터 지역 패턴을 추출하여 여러 합성곱층을 쌓아 계층적 특징을 학습한다.
- 풀링층(Pooling Layers): 합성곱층에서 추출된 특징맵의 크기를 줄여 모델의 복잡도를 낮추고 과적합을 방지한다.
- 완전 연결층(Fully Connected Layers): 추출된 특징들을 기반으로 최종 파일 유형을 분류한다.

High Entropy 값을 가진 데이터 파편들을 1D-CNN 모델로 분류한다. 이러한 파편들은 높은 무작위성을 가지기에 기존의 통계적 방법인 TF-IDF와 같은 기술로는 패턴을 식별하기 어렵기 때문이다. 이에 1D-CNN은 원시 바이트 데이터로부터 복잡하고 추상적인 특징을 자동으로 학습하는 능력이 뛰어나므로, High Entropy 값을 가진 파편 내 숨겨진 패턴을 효과적으로 탐지하여 분류할 수 있다.

5. IVI 시스템을 위한 파일 유형 분류 모델

본 논문에서 제안하는 파일 파편 유형을 분류하는 기법의 전체 구조가 그림 1에 나타나 있다. 데이터셋은 IVI 시스템에서 주로 사용되는 8개의 파일 유형들의 파편들로 구성되었다. 각 파일 유형별로 충분한 수의 파편을 수집하여 학습 및 테스트 데이터로 분리하였다. 실험에 사용한 유형별 파편 수는 표 2와 같다.

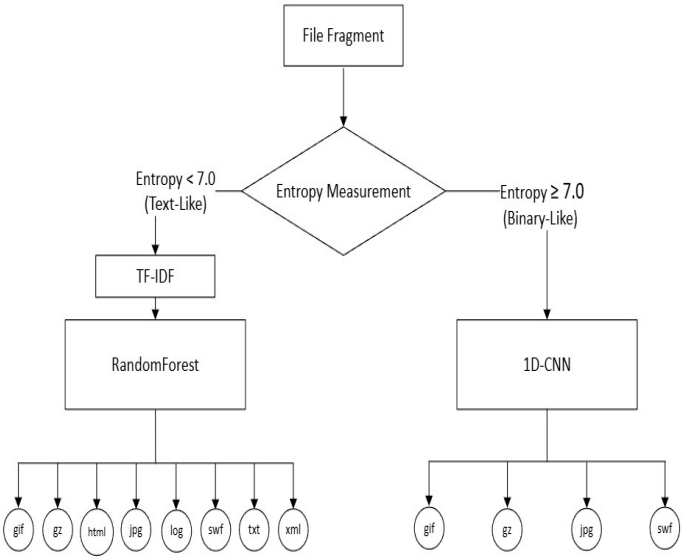

그림 1. 제안 기법의 전체 구조

표 2. 파일 유형별 학습 및 테스트용 파편 수

Type	Train	Test
gif	50,000	30,000
gz	50,000	30,000
html	50,000	30,000
jpg	50,000	30,000
log	50,000	30,000
swf	50,000	30,000
txt	50,000	30,000
xml	50,000	30,000

5.1 엔트로피 기반 분류 수행 결과

표 2에 제시된 학습과 테스트용 파편들에 대해 엔트로피를 계산하였고, 그 결과가 표 3에 나타나 있다.

엔트로피 기반 분류 결과 압축 알고리즘을 사용하는 gif, gz, jpg, swf 파일 유형들의 경우, 대부분의 파편이 High Entropy로 분류되었다. 단 일부 파편들은 Low Entropy로 분류된 경우도 확인되었다. 단 이러한 경우들은 그 개체 수가 많지 않아 학습 과정에서의 데이터 불균형 문제가 발생했기 때문일 수 있다.

표 3. 파일 유형별 파편의 엔트로피 기반 분포

Type	High Entropy		Low Entropy	
	Train	Test	Train	Test
gif	46,255	29,727	3,745	273
gz	49,639	29,850	361	150
html	28	0	49,972	30,000
jpg	49,265	29,401	735	599
log	0	0	50,000	30,000
swf	44,162	26,471	5,838	3,529
txt	0	0	50,000	30,000
xml	0	0	50,000	30,000

5.2 Low Entropy 분류 실험 결과

엔트로피 값이 7.0 미만인 Low Entropy의 경우 8개의 파일 유형 모두에서 확인되었다. Low Entropy로 분류된 파편들에 대해 TF-IDF를 특징으로 하여 RF로 학습한 모델에 대한 실험 결과를 혼동 행렬(confusion matrix)로 분석하였다 (표 4 참조).

대각선 값은 각 파일 유형에 대한 정확한 분류율을 나타내며 이외의 값은 오분류율을 나타낸다. txt 파일의 경우 96.2%의 높은 정확도로 분류되었으며, gz, html, log, swf, xml 또한 비교적 높은 분류율을 기록했다. 반면 gif, jpg는 낮은 분류율을 보였으며, gif의 경우 gz 파일로 오분류 되는 경향을 보인다. 표 3을 보면, Low Entropy에 대한 gif 파편 수는 3,745개에 불과하다. 이는 gif 파일의 경우 높은 압축률을 보이기에 대부분의 파편은 High Entropy 값을 가지지만, 일부 파편은 Low Entropy로 분류되었다. 이는 해당 샘플 수가 적어 RF 모델이 충분한 학습을 수행하지 못하였기 때문이라고 판단된다. jpg 파일 또한 동일한 이유로 낮은 성능을 기록했다.

5.3 High Entropy 분류 실험 결과

Entropy 값이 7.0 이상인 High Entropy 파일 파편에 대한 분류 결과를 표 5에 나타냈다.

표 4. Low Entropy 파일 파편에 대한 분류 성능

	gif	gz	html	jpg	log	swf	txt	xml
gif	25.7	47.3	0	0.3	0	7.9	18.8	0.1
gz	5.6	82.7	0	1.1	0	10.5	0	0
html	0.2	0.2	91.6	0	0.4	0	7.6	0
jpg	3.5	63.7	0	15.4	0.2	14.1	1.6	1.4
log	0	0	0.3	0	94.4	0	5.3	0
swf	10.4	8.2	0.4	1.7	0.1	78.7	0.5	0
txt	0	0	1.3	0	2.4	0	96.2	0
xml	0	0	1.9	0	0.2	0	2.4	95.5

표 5. High Entropy를 가진 파일 파편에 대한 분류 성능

	gif	gz	jpg	swf
gif	88.8	7.3	0.4	3.5
gz	3.1	84.2	1	11.7
jpg	1	4.9	80.9	13.1
swf	3.1	44.8	21	31.1

High Entropy를 가진 파편의 경우, 4가지의 파일 유형이 확인되었다. 일부 html 파편의 경우, 학습 데이터에는 분류되었으나 테스트 데이터에서는 분류되지 않았기에 제외한다. 실험 결과, Low Entropy를 가진 파편들에서 낮은 성능을 보였던 gif, jpg 유형에 대해 각각 88.8%, 80.9%의 정확도를 기록했다. 이는 대부분의 gif와 jpg 파편들이 High Entropy 값을 가지고 또한 충분한 파편으로 학습하였기에 안정적인 성능을 보이는 것으로 보인다. gz 파일은 84.2%의 성능을 기록하였다. swf 파일은 Entropy 계산 과정에서 대부분 High Entropy로 분류되었음에도 31.1%의 성능을 보였으며 gz 파일로 오분류(44.8%)되는 경향을 보인다. 이는 swf와 gz의 데이터 시퀀스가 서로 유사한 패턴을 갖고 있음을 의미한다.

5.4 분석 및 논의

실험을 통해, 제안하는 머신러닝 기반 파일 파편 분류 모델이 IVI 시스템 환경의 파일 유형에 대해 유의미한 분류 성능을 보인다는 것을 확인하였다. 먼저 엔트로피를 기준으로 파편을 구분하고, High Entropy를 가진 파편들은 1D-CNN으로, Low Entropy를 가진 파편들을 RF로 분류하였다.

실험 결과, gif와 jpg와 같은 무작위성 바이트 시퀀스를 갖는 유형들은 높은 분류 정확도를 보여주었다. 하지만 swf 파일의 경우 낮은 분류 정확도를 보였으며, 특정 파일 파편들에서는 오분류가 발생했다. 이는 학습 파편 수의 부족, 적합한 특징 미파악 등과 같은 다양한 원인일 수 있다. 향후 연구에서 이러한 오분류 문제를 해결하기 위한 추가적인 특징 추출 및 데이터셋 확장이 필요하다.

6. 결론 및 한계점

본 논문에서, IVI 시스템에서 파일 카빙을 위한 파일 파편 유형 분류의 필요성을 제시하고, 머신러닝 기반의 파편 분류 기법을 제안했다. 특히 IVI 시스템에서 주로 사용되는 파일 유형을 대상으로 1차 분류를 위해 엔트로피를 적용하고, 2차 분류를 위해 RF와 1D-CNN 모델을 활용하였다. 실험 결과, 제안 모델은 특정 7개의 클래스(txt, xml, log, html, gz, gif, jpg) 유형에 대해 좋은 분류 정확도를 보였다.

하지만 본 논문은 다음과 같은 한계점을 가진다. SSD가 아닌 HDD 환경을 기반으로 구축된 데이터셋을 활용했다. 대부분의 IVI 시스템은 SSD 스토리지를 사용하는 방향으로 발전하고 있다. SSD와 HDD는 근본적인 데이터 관리 방식이 다르다. SSD의 고유적 특성으로 인해 데이터 삭제 및 파편화 양상이 복잡해지며, 페이지 레벨의 미세한 파편화가 발생할 수 있다. 따라서 본 논문의 결과가 SSD 환경에 적용되지 못할 수 있으며, 실제 IVI 시스템에 적용하기 위해서는 SSD 환경에 특화된 추가 연구가 필요하다.

또한 엔트로피 기반 분류에서 데이터 불균형 문제가 발생했다. 압축률이 높은 파일 유형들의 경우 대부분의 파편이 High Entropy로 분류되지만, 일부 파편의 경우는 Low Entropy로 분류되었다. 이는 Low Entropy를 가진 파편 수의 부족으로 인해 충분한 학습이 보장되지 못했기 때문으로 추측된다.

향후 연구 방향은 다음과 같다. 실제 IVI 시스템의 SSD 환경에서 발생하는 파일 파편화 특성을 반영한 대규모 데이터셋 구축이 필요하다. 또한 현재 제안하는 모델에 대한 하이퍼파라미터 튜닝 작업을 통해 분류 정확도를 향상하는 연구가 필요하다.

Acknowledgements

이 논문은 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원-대학ICT연구센터(ITRC)의 지원(IITP-2025-RS-2023-00259967, 기여율 50%)과 산업통상자원부(MOTIE)와 한국에너지기술평가원(KETEP)의 지원을 받아 수행된 연구임(No. RS-2021-KP002461, 기여율 50%).

[참고 문헌]

[1] K. K. Sindhu and B. B. Mesharm, “Digital forensics and cyber crime datamining”, Journal of Information Security, 3, 3, pp.196-201, 2012.

[2] Z. Y. Yu, D. D. Jin, X. X. Song, C. Zhai and D. S. Wang, “Internet of Vehicle Empowerd Mobile Media Scenarios: In-Vehicle Infotainment Solutions for the Mobility as a Service (MaaS)”, Sustainability, 12, 18, 7448, 2020.

[3] Y. J. Yoon, J. H. Jung, S. J. Cho, J. M. Choi, M. K. Park and S. C. Han, “Forensic Investigation of An Android Jellybean-based Car Audio Video Navigation System”, Proceeding of the 19th International Conference on Availability, Reliability and Security, 98, pp. 1-8, 2024.

[4] R. Poisel and T. Simon Tjoa, “A comprehensive literature review of file carving”, 2013 International conference on Availability, Reliability and Security, pp. 475-484, 2013.

[5] W. Qiu, R. Zhu, J. Guo, X. Tang, B. Liu and Z. Huang, “A new approach to multimedia files

carving", IEEE International Conference on Bioinformatics and Bioengineering, pp. 105-110, 2014.
[6] H. J. Hadi, N. Musthaq and I. U. Khan, "SSD Forensic: Evidence Generation and Forensic Research on Solid State Drives Using Trim Analysis", International Conference on Cyber Warfare and Security, pp. 51-56, 2021.
[7] Z. S. Alrobieh, "File Carving Survey on Techniques, Tools and Areas of Use", Transactions on Networks and Communications, 8, 1, 2020.
[8] K. Skračić, J. Petrović and P. Pale, "Classification of low-and high-entropy file fragments using randomness measures and discrete Fourier transform coefficients", Vietnam journal of computer science, 10, 4, pp. 433-462, 2023.
[9] M. Masoumi, A. Keshavarz and R. Fotohi, "File Fragment recognition based on content and statistical features", Multimedia Tools and Applications, 80, pp. 18859-18874, 2021.
[10] M. Bhatt et al., "Hierarchy-Based File Fragment Classification" Machine Learning and Knowledge Extraction, 2, 3, pp. 216-232, 2020.
[11] K. Vulinović, L. Ivković, J. Petrović K. Skračić and P. Pale, "Neural Networks for File Fragment Classification", International Convention on Information and Communication Technology, pp. 1194-1198, 2019.
[12] 김다은, 강해인, 조성제, 김홍근, 황영섭, "안드로이드 AVN 시스템에서 머신러닝을 활용한 파일 파편의 유형 분류 기법 비교분석", 정보보안 및 고신뢰 컴퓨팅 하계워크숍, pp. 33-38, 2023.
[13] S. Garfinekl, P. Farrell, V. Roussev and G. Dinolt, "Bringing science to digital forensics with standardized forensic corpora", Digital Investigation, 6, pp. S2-S11, 2009.
[14] A. Mantovani, S. Aonzo, X. Ugarte-Pedrero, A. Merlo and D. Balzarotti, "Prevalence and impact of low-entropy packing schemes in the malware ecosystem", Network and Distributed System Security Symposium, pp. 23-26, 2020.