https://doi.org/10.33778/kcsa.2024.24.3.041

문자 인코딩 방식의 변화에 따른 트랜스포머 기반 침입탐지 모델의 탐지성능 비교

김 관 재*, 이 수 진**

요 약

트랜스포머 모델의 핵심 요소인 토크나이저는 숫자 형태의 데이터를 제대로 이해하지 못한다. 따라서 패킷 페이로드를 문장처럼 학습하여 실제 네트워크에서 동작 가능한 트랜스포머 기반의 침입탐지 모델을 구축하기 위해서는 16진수 형태의 패킷 페이로드를 문자 형태로 변환하는 것이 필요하다. 이러한 문제 인식 하에 본 연구에서는 3종의 문자 인코딩 방식을 적용하여 패킷 페이로드를 숫자 및 문자 형태로 변환한 후 트랜스포머 모델에 학습시키면서 모델의 탐지성능이 어떻게 달라지는지를 분석하였다. 성능 분석 실험을 위한 데이터세트는 UNSW-NB15 데이터세트에 포함된 PCAP 파일에서 패킷 페이로드를 추출하여 구성하였으며, 학습 모델은 RoBERTa를 사용하였다. 실험 결과, ISO-8859-1 인코딩이 이진분류 및 다중분류에서 가장 우수한 성능을 달성하는 것으로 확인되었으며, 토큰의 수를 512개로 설정하고 최대 에포크를 15회로 증가한 경우에 다중분류 정확도가 88.77%까지 향상되었다.

Performance Comparison of Transformer-based Intrusion Detection Model According to the Change of Character Encoding

Kwan-Jae Kim*, Soo-Jin Lee**

ABSTRACT

A tokenizer, which is a key component of the Transformer model, lacks the ability to effectively comprehend numerical data. Therefore, to develop a Transformer-based intrusion detection model that can operate within a real-world network environment by training packet payloads as sentences, it is necessary to convert the hexadecimal packet payloads into a character-based format. In this study, we applied three character encoding methods to convert packet payloads into numeric or character format and analyzed how detection performance changes when training them on transformer architecture. The experimental dataset was generated by extracting packet payloads from PCAP files included in the UNSW-NB15 dataset, and the RoBERTa was used as the training model. The experimental results demonstrate that the ISO-8859-1 encoding scheme achieves the highest performance in both binary and multi-class classification. In addition, when the number of tokens is set to 512 and the maximum number of epochs is set to 15, the multi-class classification accuracy is improved to 88.77%.

Key words : IDS(Intrusion Detection System), UNSW-NB15, (NLP)Natural Language Processing, Encoding

접수일(2024년 08월 12일), 수정일(1차: 2024년 09월 07일), 게재확정일(2024년 09월 27일)

* 국방대학교 국방과학학과 사이버전 석사과정(주저자)
** 국방대학교 국방과학학과 교수(교신저자)

1. 서 론

문장 의미 파악이나 새로운 문장의 생성, 번역 등 자연어 처리에서 강력한 성능을 발휘하는 트랜스포머 모델은 음성 인식 및 이미지 처리 등 다른 분야에서도 뛰어난 성능을 보여주고 있어 응용 분야가 점점 더 넓어지고 있다. 최근에는 트랜스포머 모델이 제공하는 정교한 Attention 메커니즘을 통해 입력 특성과 다양한 침입 유형 간의 관계를 보다 통찰력 있게 분석할 수 있다는 장점을 활용하여 침입탐지 분야에서의 적용도 증가하고 있다.

트랜스포머 모델은 침입탐지 모델 구축과 관련하여 우리에게 새로운 가능성을 제시해 준다. 선행연구를 통해 제시된 일반적인 인공지능 기반 침입탐지 모델들은 패킷을 분석하여 가공한 후 생성되는 합성 데이터인 메타데이터(meta-data)를 이용해 학습을 수행한다. 따라서 메타데이터와 동일한 통계적 특성을 가지는 테스트 데이터에 대해서는 거의 완벽에 가까운 탐지성능을 보인다. 그러나 통계적 특성이 다른 데이터, 예를 들어 네트워크로 유입되는 실제 패킷을 기반으로 한 침입탐지에서도 똑같은 성능을 보장한다고 확신하기는 어렵다. 그러나 트랜스포머 모델을 적용할 경우에는 수집된 침입 패킷을 문장처럼 학습할 수 있어 실제 네트워크에서도 동작이 가능한 침입탐지 모델 구축이 가능해진다. 이러한 점에 착안하여 본 연구진은 BERT 및 DistilBERT 2종의 트랜스포머 모델을 기반으로 UNSW-NB15 데이터세트에서 제공하는 PCAP 파일의 패킷 페이로드를 학습시켜 침입을 탐지하는 모델을 제안한 바 있으며, 선행연구 대비 향상된 탐지성능을 달성할 수 있음도 확인하였다[1].

한편 트랜스포머 모델의 자연어 처리 성능을 좌우하는 핵심 요소는 토큰화(tokenization)라고 할 수 있다. 토큰화는 처리 대상이 되는 자연어 문장을 문자, 단어 또는 서브워드(subword) 단위로 분절하여 트랜스포머 모델이 입력으로 사용할 토큰 시퀀스(token sequence)를 생성한다. 이 과정에서 토크나이저가 이전에 학습하지 않은 데이터를 대상으로 토큰화를 수행하는 경우 중대한 오류가 발생할 수 있다. 영어가 아닌 언어에 대해서는 토큰화가 제대로 수행되지 않아 언어모델의 성능이 저하되는 현상이 대표적인 사례이다. 그리고 숫자들이 반복해서 출현하면 특정 숫자가 가지는 의미를 문맥 속에서 이해하지 못하여 그 숫자들을 의미 없는 방식으로 토큰화할 수 있다[2].

이 외에도 현재 수준의 토크나이저가 가지고 있는 한계는 다양하지만 본 연구에서는 숫자 형태의 데이터를 제대로 이해하지 못하는 문제에 집중하고자 한다. 전술한 바와 같이 본 연구진이 수행한 선행연구에서는 16진수 형태의 패킷 페이로드를 전처리 없이 문장 형태로 학습하였다. 그러나 문자 인코딩(character encoding)을 적용하여 16진수가 아닌 다른 형태로 변환하고 토큰화가 수행될 경우에 탐지성능이 달라질 수 있는지에 대해서는 확인하지 못하였다. 이에 본 논문에서는 패킷 페이로드에 3종의 대표적인 문자 인코딩을 적용하여 트랜스포머 모델에 학습시킨 후 탐지성능의 변화를 분석한다.

본 논문의 구성은 다음과 같다. 2장에서는 본 연구에서 사용한 UNSW-NB15 데이터세트와 트랜스포머 모델들에 대해 살펴보고, 동일한 데이터세트를 사용했던 선행연구를 정리한다. 3장에서는 제안된 침입탐지 모델의 구축 절차와 실험 방법을 설명한 후, 결과를 분석한다. 마지막으로 4장에서 연구를 요약하고 결론을 맺는다.

2. 데이터세트 및 모델

2.1 UNSW-NB15 Dataset

UNSW-NB15 데이터세트[3, 4]는 호주 사이버보안센터(ACCS)의 Cyber Range Lab에서 제작한 데이터세트로, ‘KDDCUP 99’[5], ‘NSLKDD’[6] 데이터세트의 중복 및 불균형 등 문제를 개선하였고, 공개 이후 현재까지도 다방면의 사이버보안 연구에서 꾸준하게 활용되고 있다.

‘IXIA PerfectStorm’을 활용하여 정상 트래픽 및 공격 트래픽을 생성하였으며, 해당 네트워크의 패킷을 캡쳐해 생성한 PCAP 파일을 ‘Argus’와 ‘Bro-IDS Tool’로 분석하여 특성을 추출한 후 정상 및 비정상 트래픽으로 분류하였다. CSV 파일과 PCAP 파일이 함께 제공되며, 총 9종의 공격에 대한 트래픽을 포함하고 있다. CSV 파일 데이터세트는 175,341개의 학습 데이터와 82,232개의 테스트 데이터로 구분되어 있으

며, 정상 트래픽 및 공격 유형별 데이터의 세부 현황은 <표 1>에서 보는 바와 같다.

본 연구는 트랜스포머 모델을 기반으로 실제 네트워크에서 동작 가능한 침입탐지 모델을 구축하는 것이 주목적이기 때문에, CSV 파일로 제공되는 메타데이터가 아니라 PCAP 파일을 모델 학습 및 평가에 사용하였다. 그러나 PCAP 파일은 라벨링이 되어 있지 않아 전처리 과정에서 CSV 파일을 사용하여 라벨링을 수행하였다.

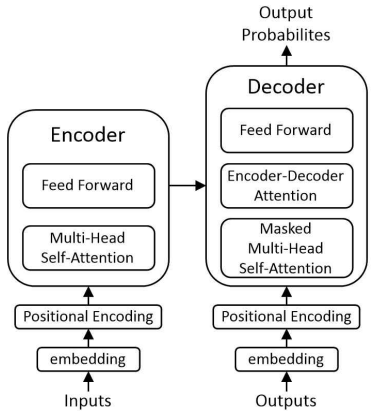
<표 1> UNSW-NB15 Dataset 구성

Cetegory	Train	Test
Analysis	2,000	677
Backdoor	1,746	583
Dos	12,264	4,089
Exploits	33,393	11,132
Fuzzers	18,185	6,062
Generic	40,000	18,871
Normal	56,000	37,005
Reconnaissance	10,492	3,496
Shellcode	1,133	378
Worms	130	44
Total	175,343	82,337

2.2 Transformer Model

트랜스포머 모델은 자연어 처리 분야에서 혁신적인 역할을 한 딥러닝 모델로서, 2017년 발표된 논문 "Attention is All You Need"에서 처음 소개되었다[7]. 이 모델은 입력을 고차원 벡터로 변환하는 인코더(Encoder)와 벡터를 통해 출력문장을 생성하는 디코더(Decoder)로 구성되어 있고, 구조는 (그림 1)에서 보는 바와 같다. 입력은 병렬적으로 처리되며, 임베딩(embedding)을 통해 각 단어는 벡터로 변환된다. 이어서 포지셔널 인코딩(Positional Encoding)을 통해 각 단어의 위치정보를 인식한다. 인코더와 디코더는 Attention 메커니즘을 통해 문장 내 각 단어가 다른 모든 단어와 가지는 관계를 동시에 학습한다.

트랜스포머 모델이 등장한 이후, 최근에는 다양한 후속 트랜스포머 모델들이 개발되고 있다. 대표적으로 BERT와 BERT 기반의 파생 모델인 DistilBERT, RoBERTa 등이 있으며, OpenAI의 GPT와 Meta의 Llama도 트랜스포머 모델의 한 종류이다.



(그림 1) Transformer 모델 구조

BERT는 2018년 구글에서 발표한 모델로서, 다양한 장르와 주제의 도서 데이터세트인 BookCorpus (6GB)와 영문 위키디피아(16GB)를 학습 데이터세트로 사용하였다. 그리고 입력 문장의 전후 문맥을 동시에 학습하는 양방향 트랜스포머 인코더 구조를 사용하여 이전 모델들보다 우수한 성능을 달성하였다[8].

DistilBERT 모델은 BERT 모델을 경량화한 버전으로, 학습 데이터세트는 BERT와 동일하다. 그러나 BERT 모델보다 40% 적은 약 66M의 파라미터를 가지고서도 성능은 BERT 모델의 97% 수준을 유지하였다[9].

RoBERTa 모델은 BERT 모델의 개선 버전으로, BERT 모델 대비 약 10배의 데이터를 사용해 다양한 언어 표현과 문맥을 학습하였고, 동적 마스킹 기법과 하이퍼파라미터 최적화를 통해 성능을 개선하였다[10]. 학습 데이터세트는 BERT와 DistilBERT에서 사용한 데이터세트에 더해 뉴스 기사 데이터세트(Common Crawl News, 76GB), 웹 페이지 텍스트 데이터세트(OpenWebText, 38GB) 및 이야기 형식의 데이터세트(STORIES, 31GB)를 추가로 학습하였다.

본 연구에서는 다양한 트랜스포머 모델 중 다양한 언어 표현과 문맥을 학습하고 성능까지 개선된 RoBERTa 모델을 사용하였다.

2.3 관련연구

본 절에서는 UNSW-NB15 데이터세트를 사용하여 침입탐지 모델을 구축한 선행연구들을 살펴보고, 이후

트랜스포머 모델을 사용한 사례들을 정리한다.

Jing 등[11]은 UNSW-NB15 원본 데이터세트에 로그함수를 사용한 비선형 스케일링으로 데이터를 전처리하는 방식을 제안하였다. 모델은 기계학습 모델인 Support Vector Machine(SVM)을 사용하였으며, 이진분류에서 85.99%, 다중분류에서 75.77%의 성능을 달성하였다.

Meftah 등[12]은 이진분류 후 다중분류를 실시하는 2단계 접근방식을 제안하였다. 모델은 SVM, 로지스틱 회귀, 결정트리 등을 사용하였으며, 학습데이터는 Random Forest 모델로 주요 특성을 선택하여 적용하였다. 이진분류는 SVM 모델이 82.11%로 가장 높은 성능을 나타냈으며, 다중분류는 SVM과 결정트리를 조합했을 때 86.04%의 성능을 달성하였다.

Kasongo 등[13]은 XGBoost 모델을 사용하여 데이터세트의 특성 중요도를 계산하는 방식을 제안하였다. 학습 및 평가에는 인공신경망, 로지스틱 회귀 모델 등을 사용하였으며, 이진분류 정확도는 결정트리를 사용했을 때 90.85%, 다중분류에서는 인공신경망을 사용했을 때 77.51%로 가장 높게 나타났다.

Bagui 등[14]은 랜덤 샘플링(random sampling), SMOTE 및 ADASYN을 각각 및 조합하여, 샘플링을 통한 불균형 데이터의 성능 개선 효과를 비교하였다. 학습 및 평가에는 인공신경망 모델을 사용하였으며, 샘플링 전보다 36.36% 향상된 77.58%의 재현율(recall)을 보여줌을 확인하였다.

Yin 등[15]은 IG(Information Gain)와 RF(Random Forest)를 조합하여 적용한 이후, RFE(Recursive Feature Elimination)를 적용하는 방식으로 2단계에 걸쳐 특성을 선택하는 접근방법을 제안하였다. 전처리 과정에서는 소수클래스 제거 및 오버샘플링을 수행했으며, 학습에는 다층퍼셉트론(Multi-layer perceptron)을 사용하여 84.24%의 다중분류 성능을 달성하였다.

상기 연구들은 UNSW-NB15 데이터세트에서 제공하는 메타데이터만을 대상으로 다양한 학습 모델과 데이터처리 기법을 적용하여 침입탐지 모델의 성능향상을 도모하였다. 그러나 본 연구는 PCAP 파일에서 패킷 페이로드를 직접 추출하여 라벨링을 실시한 후 데이터세트로 활용했고, 전처리 과정에서 문자 인코딩 방식을 적용했다는 점에서 선행연구들과 차별화된다. 선행연구에서 적용된 학습 모델 및 데이터처리 기법들을 종합적으로 비교하여 정리한 결과는 <표 7>에서 확인할 수 있다.

한편, 본 연구와 유사하게 트랜스포머 모델을 사용하여 침입탐지 모델을 구축하고자 시도한 사례들도 있다. Wu 등[16]은 트랜스포머 모델의 디코더에 Self Attention 하위 계층을 하나 더 추가한 변형 트랜스포머를 사용하여 침입탐지 모델을 구축하였다. 학습 데이터는 CICDS2017와 CIC-DDoS 2019 데이터세트를 사용하였으며, 이진분류 정확도는 99.98%와 99.65%로 나타났다.

Yang 등[17]은 트랜스포머 모델을 이미지 인식 분야에 적용한 ViT(Vision Transformer) 모델을 사용하여 침입탐지 모델을 구축하였다. 학습 데이터는 NSL-KDD 데이터세트를 사용했으며, 이진분류에서 99.68%의 성능을 달성하였다.

서론에서 언급한 본 연구진의 선행연구[1]에서는 전처리 과정을 생략하기 위한 목적으로, 패킷의 페이로드를 16진수 형태의 한 문장으로 처리하는 방법을 제안하였다. 학습 및 평가에는 BERT와 DistilBERT 모델을 사용하였으며, 이진분류에서 99.03%, 99.05%의 정확도를 보여주었으며, 다중분류에서는 86.63%, 86.36%의 정확도를 달성하였다.

3. 제안하는 방법

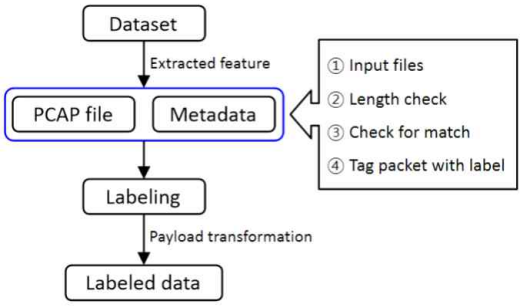
3.1 실험 준비

3.1.1 데이터세트 라벨링

UNSW-NB15의 PCAP 파일은 이진(binary) 파일 형식이고 라벨링이 되어있지 않아 지도학습에 그대로 사용하기는 제한이 있다. 따라서, PCAP 파일에서 페이로드를 추출하고 정상과 공격, 공격 유형을 라벨링하는 과정을 선행하였고, 이 과정에는 Farrukh 등이 제안한 'Payload- Byte' 기법[18]을 사용하였다.

Payload-Byte 기법은 PCAP 파일 안의 각 패킷의 헤더와 CSV 파일의 메타데이터를 매칭하여 서로 일치하는 패킷을 찾고, 해당 패킷의 페이로드에 라벨링을 실시한 뒤 CSV 파일로 출력한다. 이 과정에서 IP, Port, 패킷의 생존시간(TTL) 등 8개의 특성을 사용하

며, 세부 동작 과정은 (그림 2)에서 보는 바와 같다.



(그림 2) Payload-Byte 동작 개념

Payload-Byte 기법으로 라벨링을 완료한 뒤에는 결측치가 포함되거나 중복된 인스턴스(instances)를 제거하였다. Normal 클래스는 랜덤언더샘플링을 통해 전체 데이터세트에서 차지하는 비중을 조정하였으며, 모든 과정을 거친 후 실제 실험에 사용된 데이터세트는 <표 2>에서 보는 바와 같다.

<표 2> 실험 데이터세트 구성

Class	Train	Test
Analysis	538	134
Backdoor	620	155
Dos	2,210	552
Exploits	9,794	2,449
Fuzzers	8,476	2,119
Generic	9,457	2,364
Normal	9,849	2,463
Reconnaissance	5,644	1,411
Shellcode	680	170
Worms	66	17
Total	47,334	11,834

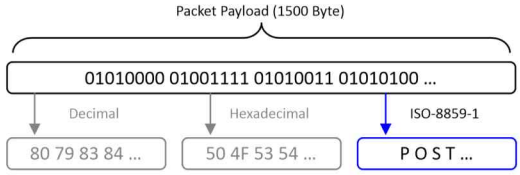
3.1.2 ISO-8859-1(Latin-1)

최초의 문자 인코딩 방식 중 하나로는 ASCII가 있다. 이 방식은 7비트로 구성된 인코딩 방식이며, 단일 언어로 영어만 표현이 가능하다.

출력이 불가능한 제어문자 33개와 출력 가능한 95개의 문자 등 총 128개의 문자를 표현할 수 있다[19].

반면, ISO-8859-1 인코딩은 ASCII와 호환이 가능하고 영어, 독일어, 프랑스어 등 주요 유럽 언어들의 문자도 표현할 수 있다. 8비트(1바이트) 고정 인코딩 방식이며, 제어문자와 공백, 특수문자 등을 모두 포함하면 총 256개의 문자를 표현할 수 있다. 또한, 1바이트를 문자로 변환하는데 발생하는 손실이 ASCII 보다 적어 1바이트를 문자로 변환하는데 적합하다[20].

UNSW-NB15 데이터세트에 Payload-Byte 기법을 적용하여 페이로드 추출 및 라벨링 과정을 거치면, 페이로드의 바이트열은 각각의 바이트가 하나의 특성이 되어 10진수 형태의 CSV 파일로 저장된다. 본 연구진의 선행연구[1]에서는 패킷 페이로드의 전처리 과정을 간소화하면서 그대로 문장처럼 학습하는 것이 주목적이었기 때문에 바이트열을 16진수 형태로 재변환하여 학습하였다. 그러나 본 연구에서는 10진수 형태의 패킷 페이로드를 16진수, ASCII 및 ISO-8859-1 인코딩 방식으로 디코딩하여 학습한 후 탐지성능을 비교하였으며, 3종의 인코딩 방식 중 ISO-8859-1 인코딩에 의한 변환 결과는 (그림 3)에서 보는 바와 같다.



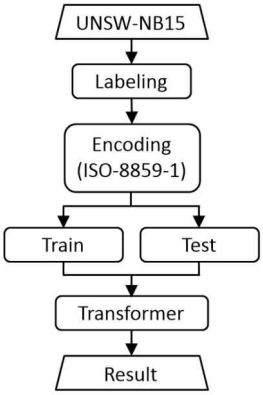
(그림 3) ISO-8859-1을 통한 변환 결과

3.2 실험 설계

Payload-Byte 기법으로 원본 PCAP 파일에서 추출된 패킷 페이로드를 3종의 문자 인코딩 방식을 통해 문자 형태로 재변환한 결과, ASCII 및 ISO-8859-1 인코딩의 경우에는 최대 1,500자로 구성된 하나의 문장으로 변환되었다. 선행연구[1]와 동일하게 16진수 형태로 변환했을 때에는 문장의 길이가 3,000~4,500자로 나타났다. 즉 ASCII 및 ISO-8859-1 인코딩을 적용하면 16진수로 변환했을 때와 비교해 문장의 길이가 절반 이하로 축소된다.

라벨링과 문자 형태로의 재변환이 완료된 데이터세트는 8대 2의 비율로 학습 데이터세트와 테스트 데이터세트로 구분하여 학습과 테스트를 진행하였다. 학습 모델은 RoBERTa 모델을 사용하였으며, 선행연구와 동일한 조건으로 최대 입력 길이(토큰 수) 128개, 에포크(epoch) 3, 배치 크기(batch size) 32, 학습률

(learning rate) 2e-5로 설정하고 각 10회 반복 실험하였다. 세부적인 실험과정은 (그림 4)에서 보는 바와 같고, 실험은 구글에서 제공하는 Colab pro 환경(python 3.10, L4 GPU, 53GB RAM)에서 진행되었다.



(그림 4) 실험 설계

3.3 결과 및 분석

문자 인코딩 방식별 이진분류 성능은 <표 3>에서 보는 바와 같다. 3종의 인코딩 방식 중 ISO-8859-1 인코딩을 적용했을 때의 탐지성능이 가장 우수하게 나타났다. 본 연구진의 선행연구 결과와 비교했을 때에도 ISO-8859-1 인코딩의 적용이 미세하지만 성능 향상 효과를 가져올 수 있음을 알 수 있다.

<표 3> 문자 인코딩 방식에 따른 이진분류 성능 비교

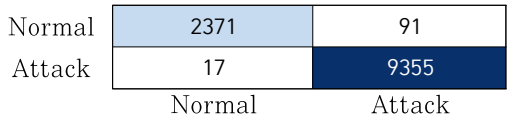
Encoding	Accuracy	F1-score	Precision	Recall
ISO-8859-1	99.15	99.47	99.08	99.86
ASCII	99.15	99.46	99.05	99.88
Hex	99.07	99.42	98.96	99.87

다중분류는 Macro F1-score를 사용하여 평가하였으며, <표 4>에서 보는 것처럼 이진분류와 동일하게 ISO-8859-1 인코딩을 적용했을 때 탐지성능이 가장 우수하였다. 그리고 16진수 형태로 학습했던 선행연구보다 모든 성능평가 지표가 향상되어 문자 인코딩의 적용이 성능향상에 큰 영향을 미침을 확인하였다.

<표 4> 문자 인코딩 방식에 따른 다중분류 성능 비교

Encoding	Accuracy	F1-score	Precision	Recall
ISO-8859-1	87.35	70.54	82.04	68.25
ASCII	87.25	70.33	80.21	68.56
Hex	86.49	64.46	70.08	63.58

이진분류 혼동행렬은 (그림 5)에서 보는 바와 같다. Normal 클래스가 Attack 클래스로 오분류되는 경우가 조금 더 많이 발생하였고, 이러한 현상은 Normal 클래스를 언더샘플링하면서 발생한 결과로 판단된다. 실제 Normal 클래스에 대한 언더샘플링을 수행하지 않고 실험을 진행했을 때에는 Normal 클래스에 대한 오분류가 거의 발생하지 않았기 때문이다. 그러나 Attack 클래스를 Normal 클래스로 오분류하는 경우가 증가하여 전체적인 탐지성능은 저하되었다.



(그림 5) 이진분류 혼동행렬

다중분류 혼동행렬은 (그림 6)에서 보는 바와 같다.

	①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩
①	11	0	2	118	0	0	0	3	0	0
②	0	13	10	111	1	1	0	17	2	0
③	1	0	141	365	18	5	0	20	2	0
④	11	1	49	2241	86	22	12	24	3	0
⑤	1	0	12	133	1936	11	5	17	4	0
⑥	1	0	13	145	12	2173	4	18	0	0
⑦	0	0	6	22	68	1	2366	0	0	0
⑧	0	1	13	117	1	0	0	1279	0	0
⑨	0	0	0	0	0	0	0	0	170	0
⑩	0	0	0	0	1	0	0	0	0	16

① Analysis ② Backdoor ③ Dos
④ Exploit ⑤ Fuzzers ⑥ Generic
⑦ Normal ⑧ Reconnaissance
⑨ Shellcode ⑩ Worms

(그림 6) 다중분류 혼동행렬

Attack에 속하는 클래스 중 데이터 수가 가장 적은 Worms(⑩)의 경우 비교적 정확하게 탐지되었지만,

그 다음으로 데이터의 수가 적은 Analysis(①)와 Backdoor(②)의 경우 대부분 다른 Attack 클래스로 오분류가 되었다. 그리고 오분류가 집중되는 클래스는 Attack 클래스 중 데이터의 수가 가장 많은 Exploit (④)으로 나타났다.

이어지는 실험에서는 토큰의 수 및 최대 에포크의 변화가 다중분류 성능에 미치는 영향을 분석하였다. 먼저 토큰 수만 128개에서 512개로 늘리고 다른 실험 조건은 동일하게 설정한 경우에는 <표 5>에서 보는 바와 같이 성능향상 효과가 크지 않았다. 그러나 토큰 수를 512개로 늘리고 최대 에포크까지 15회로 증가시킨 경우에는 <표 6>에서 보는 바와 같이 다중분류 성능이 보다 큰 폭으로 향상되었다. 이러한 추가 실험은 Colab pro+ 환경(python 3.10, A100 GPU, 83.5GB RAM)에서 진행되었다.

<표 5> 토큰 수 변화(128→512)에 따른 다중분류 성능

Encoding	Accuracy	F1-score	Precision	Recall
ISO-8859-1	87.47	69.76	81.36	68.29
ASCII	87.32	69.69	79.05	68.13
Hex	86.82	66.41	75.69	65.32

<표 6> 토큰 수 및 에포크 변화에 따른 다중분류 성능

Encoding	Accuracy	F1-score	Precision	Recall
ISO-8859-1	88.77	76.06	84.06	76.07
ASCII	88.74	75.51	84.78	75.78
Hex	88.40	75.50	84.77	77.56

이상과 같은 실험 결과를 기초로 가장 높은 성능을 달성한 ISO-8859-1 인코딩 기반 모델의 성능과 선행연구에서 제시된 성능을 비교한 결과는 <표 7>에서 보는 바와 같다. 이진분류 성능은 본 연구진의 선행연구와 동일한 실험조건 하에서 달성한 성능이며, 다중분류 성능은 ISO-8859-1 인코딩을 적용하고 토큰의 수는 512개, 최대 에포크는 15회로 설정하여 달성한 최고 성능을 기준으로 비교하였다.

그 결과 본 연구에서 평가한 ISO-8859-1 인코딩 방식 및 RoBERTa 모델을 조합한 접근방법이 가장 우수한 탐지성능을 달성하였다. 이진분류 성능은 본 연구진의 선행연구에서 달성한 성과 대비 크게 향상되지는 않았으나 다중분류 측면에서는 정확도가 2.14%p 향상되었다. 그러나 희소 클래스 등 데이터 불균형 문제를 고려하지 않아 데이터 수가 적은 클래스에서 다소의 오분류가 발생하여 전체적인 정확도를 제외하고는 선행연구 대비 큰 성능향상을 달성하지 못하였다.

4. 결 론

침입탐지 분야에서도 활용이 증가하고 있는 트랜스포머 모델은 토큰화 과정에서 숫자 형태의 데이터를 제대로 이해하지 못한다는 한계를 가지고 있다. 이에 본 연구에서는 패킷 페이로드를 문장 형태로 학습할 수 있는 트랜스포머 기반 침입탐지 모델을 구축함에 있어 문자 인코딩을 적용하여 16진수 형태의 페이로

<표 7> 이진분류 및 다중분류 성능 비교 결과

	Research	Main Approach	Accuracy	F1-score	Precision	Recall
이진 분류	Ours	ISO-8859-1 Encoding & RoBERTa	99.15	99.47	99.08	99.86
	[1]	Hex Encoding & BERT/DistilBERT	99.03	99.39	99.04	99.74
	[11]	특성 선택 & SVM	85.99	-	-	-
	[12]	Support Vector Machine	82.11	-	-	-
	[13]	Decision Tree	90.85	88.45	80.33	98.38
다중 분류	Ours	ISO-8859-1 Encoding & RoBERTa	88.77	76.06	84.06	76.07
	[1]	Hex Encoding & BERT/DistilBERT	86.63	69.75	80.71	68.94
	[11]	특성 선택 & SVM	75.77	-	-	-
	[12]	이진분류 후 다중분류 & SVM+DT	86.04	-	-	-
	[13]	특성 선택 & 인공신경망	77.51	77.28	79.50	77.53
	[14]	샘플링 & 인공신경망	-	44.36	40.00	77.58
	[15]	특성 선택 & 오버샘플링 & 다층퍼셉트론	84.24	82.85	83.60	84.24

드를 문자 형태로 변환하고 탐지성능에 미치는 영향을 분석하였다.

UNSW-NB15 데이터세트에 포함되어 있는 PCAP 파일을 사용하여 Payload-Byte 기법을 통해 패킷 페이로드를 추출하고 라벨링을 수행한 후 중복값 등을 제거하고 실험에 사용할 데이터세트를 구축하였다. 학습 모델은 RoBERTa를 사용하였으며, 문자 인코딩은 16진수, ASCII 및 ISO-8859-1 3종을 적용하여 탐지성능을 비교하였다.

그 결과 ISO-8859-1 인코딩 방식을 통해 숫자를 문자 형태로 변환하여 모델에 입력할 경우 가장 높은 성능을 보였다. 이진분류에서는 Accuracy 99.15% 및 F1-score 99.47%, 다중분류에서는 Accuracy 87.35% 및 F1-score 70.54%를 달성하였다. 또한, 본 연구에서 제안하는 접근방법이 달성 가능한 최고 성능을 확인하기 위해 토큰의 수를 최대인 512개로 증가시키고 최대 에포크도 15회로 변경하여 ISO-8859-1 인코딩 방식을 적용하면 다중분류 Accuracy를 88.77%까지 향상시킬 수 있었다. 이러한 결과를 통해 트랜스포머 기반 침입탐지 모델을 구축하는 경우 모델에 입력될 데이터의 형태를 반드시 고려해야 한다는 점을 알 수 있다. 토크나이저가 제대로 이해하지 못하는 숫자 형태의 데이터보다는 문자 형태의 데이터로 변환하여 입력하는 것이 보다 바람직한 접근방법이 될 것이다.

한편 문자 인코딩의 적용을 통해 트랜스포머 기반 침입탐지 모델의 성능이 전체적인 정확도 측면에서 개선될 수 있음은 확인하였지만, 희소 클래스 등 학습 데이터의 불균형은 간과하여 다중분류에서 다수의 오분류가 발생하는 문제는 해결하지 못하였다. 따라서 향후 연구에서는 SMOTE나 ADASYN 등의 샘플링 기법을 적용하여 전반적인 성능을 개선하는 연구를 진행할 예정이다.

참고문헌

[1] Woo-Seung Park, Gun-Nam Kim, and Soo-Jin Lee, “Intrusion Detection System based on Packet Payload Analysis using Transformer”, Journal of the Korea Society of Computer and Information , 28(11), 81-87, 2023.

[2] D. Spathis and F. Kawsar, “The first step is the hardest: pitfalls of representing and tokenizing temporal data for large language models”, Journal of the American Medical Informatics Association, Jul 2024.

[3] N. Moustafa and J. Slay, “UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)”, Military Communications and Information Systems Conference (MilCIS 2015), pp. 1-6, Nov 2015.

[4] Canadian Institute for Cybersecurity, “UNSW-NB15 Data set”, https://www.unsw.adfa.edu.au/unsw-canberra-cyber/cybersecurity/ADFA-NB15-Datasets/

[5] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, “A Detailed Analysis of the KDD CUP 99 Data Set”, Proceedings of the IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA 2009), pp. 1-6, July 2009.

[6] Australian Center for Cyber Security, “NSL-KDD dataset”, https://www.unb.ca/cic/datasets/nsl.html

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is All You Need”, Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS 2017), pp. 6000-6010, Dec 2017.

[8] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language

Understanding", 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1, pp. 4171-4186, May 2019.

[9] V. Sanh, L. Debut, J. Chaumond and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter", Mar 2020.

[10] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A Robustly Optimized BERT Pretraining Approach", arXiv:1907.11692, 2019.

[11] D. Jing and H. B. Chen, "SVM Based Network Intrusion Detection for the UNSW-NB15 Dataset", 2019 IEEE 13th International Conference on ASIC (ASICON), Chongqing, China, pp. 1-4, 2019.

[12] S. Meftah, T. Rachidi, and N. Assem, "Network Based Intrusion Detection Using the UNSW-NB15 Dataset", International Journal of Computing and Digital Systems, Vol. 8(5), 2019.

[13] S. M. Kasongo and Y. Sun, "Performance Analysis of Intrusion Detection Systems Using a Feature Selection Method on the UNSW-NB15 Dataset", Journal of Big Data 7(105), Nov. 2020.

[14] S. Bagui and K. Li, "Resampling imbalanced data for network intrusion detection datasets", Journal of Big Data 8(6), Jan. 2021.

[15] Y. Yin, J. Jang-Jaccard, W. Xu, A.Singh, and J. Zhu, "IGRF-RFE: a hybrid feature selection method for MLP-based network intrusion detection on UNSW-NB15 dataset", J Big Data 10, 15, 2023.

[16] Z. Wu, H. Zhang, P. Wang and Z. Sun, "RTIDS: A Robust Transformer-Based Approach for Intrusion Detection System", IEEE Access, vol. 10, pp. 64375-64387, 2022.

[17] Y-G. Yang, H-M. Fu, S. Gao, Y-H. Zhou, and W-M. Shi, "Intrusion detection: A model based on the improved vision transformer", Transactions on Emerging Telecommunications Technologies 33(9):e4522, Apr. 2022.

[18] Y. A. Farrukh, I. Khan, S. Wali, D. Bierbrauer, and N. Bastian, "Payload-Byte: A Tool for Extracting and Labeling Packet Capture Files of Modern Network Intrusion Detection Datasets,", pp 58-67, Sep. 2022.

[19] Charles E. Mackenzie, 'Coded Character Sets, History and Development', Addison-Wesley Publishing Company, Inc., 1980.

[20] 'ISO/IEC 8859-1:1998', ISO, 1998.

〔저자소개〕

김 관 재 (Kwan-Jae Kim)
2013년 2월 가톨릭대학교 정보통신전자공학부 학사
2023년 ~ 현재 국방대학교 국방과학학과 사이버전 석사과정
email : blacknwhite-cyber@naver.com

이 수 진 (Soo-Jin Lee)
1992년 3월 육군사관학교 전산학과 (이학사)
1996년 2월 연세대학교 컴퓨터과학과 (공학석사)
2006년 2월 한국과학기술원 전산학과 (공학박사)
2006년 3월 ~ 현재 국방대학교 국방과학학과 컴퓨터공학/사이버전 교수
email : cyberkma@korea.kr