https://doi.org/10.7840/kics.2025.50.2.304

비암호화 데이터 학습을 통한 응용 트래픽 분류

김 주 성◆, 장 윤 성*, 백 의 준*, 김 명 섭°

Application Traffic Classification through Non-Encrypted Data Learning

Ju-Sung Kim◆, Yoon-Seong Jang*, Ui-Jun Baek*, Myung-Sup Kim°

요 약

인터넷의 성장과 더불어 다양한 응용 트래픽이 등장하고 정보보호의 중요성이 점점 더 강조되고 있다. 이에 따라 암호화 통신의 사용이 증가하고 있으며, 이러한 환경에서 효과적인 네트워크 트래픽 분류의 필요성이 커지고 있다. 그러나 기존의 전통적인 네트워크 트래픽 분류 기법은 암호화된 데이터나 복잡한 패턴을 다루는데 한계를 보였고 이를 극복하기 위해 딥러닝 기반의 새로운 분류 방법이 등장하였다. 하지만 딥러닝 기반 분류 방법 역시 암호화된 데이터나 노이즈가 많은 환경에서는 낮은 성능을 보인다는 문제점이 있다. 따라서 본 논문에서는 이에 대응하기 위해, TLS 헤더 값을 기준으로 트래픽 데이터를 암호화 여부에 따라 구분하고, 암호화되지 않은 데이터만을 딥러닝 모델의 학습 데이터로 사용하는 방법론을 제안한다. 제안한 방법론을 적용하여 실험을 진행하였으며 일반적인 전처리 방법보다 평균 9%p 높은 정확도를 보였다.

키워드 : 응용 트래픽, 트래픽 분류, 딥러닝
Key Words : Application Traffic, Traffic Classification, Deep Learning

ABSTRACT

With the growth of the internet, various application traffic types have emerged, and the importance of information security is increasingly emphasized. Consequently, the use of encrypted communication has risen, heightening the need for effective network traffic classification in such environments. However, traditional network traffic classification techniques have shown limitations in handling encrypted data and complex patterns, leading to the emergence of new deep learning-based classification methods. Despite this, deep learning-based classification methods also face challenges in environments with encrypted data or high levels of noise, resulting in lower performance. To address these issues, this paper proposes a methodology that classifies traffic data based on TLS header values to distinguish between encrypted and unencrypted data and utilizes only unencrypted data as training input for the deep learning model. The experiment was conducted using the proposed methodology, resulting in an average accuracy that was 9 percentage points higher than that of the general preprocessing method.

※ 본 논문은 2024년도 교육부의 재원으로 한국연구재단의 지원을 받아 수행된 지자체-대학 협력 기반 지역혁신 사업(2021RIS-004)과 2023년도 정부(산업통상자원부)의 재원으로 한국산업기술진흥원의 지원(P0024177, 2023년 지역혁신클러스터육성)을 받아 수행된 연구임.
◆ First Author : Korea University Department of Computer and Information Science, jsung0514@korea.ac.kr, 학생회원
° Corresponding Author : Korea University Department of Computer and Information Science, tmskim@korea.ac.kr, 종신회원
* Korea University Department of Computer and Information Science, brave1094@korea.ac.kr, 학생회원; pb5846@korea.ac.kr
논문번호 : 202409-193-C-RN, Received September 3, 2024; Revised September 29, 2024; Accepted October 31, 2024

Ⅰ. 서 론

인터넷의 성장은 다양한 온라인 서비스와 응용 프로그램의 등장을 가져왔고, 이에 따라 네트워크 상에서 교환되는 데이터의 유형과 양상도 다양해졌다. 이러한 다양성은 기존의 트래픽 관리 전략에 새로운 도전을 제기하고 있으며, 트래픽의 효과적인 분류와 관리는 네트워크 성능 최적화와 보안 유지에 필수적이다. 특히, 트래픽의 급증은 네트워크 자원의 효율적 사용, 서비스 품질 보장 및 네트워크 보안 위협에 대응하기 위해 더욱 정밀한 트래픽 분류 기법의 개발을 요구한다.

응용 트래픽 분류는 네트워크에서 전송되는 데이터 트래픽을 다양한 응용 프로그램이나 서비스 유형에 따라 구분하는 작업이다. 응용 트래픽을 분류하기 위한 방법으로 포트 기반 분류, 페이로드 기반 분류, 머신러닝 기반 분류, 딥러닝 기반 분류가 있다[1]. 포트 기반 분류나 페이로드 기반 분류 방법은 구현이 간단하고 분류 속도가 빠르다는 장점이 있다. 하지만 정보보호의 중요성이 높아짐에 따라 데이터 암호화 비율도 상승하는 추세이며, 대부분의 응용에서 기존의 평문 통신 대신 암호화 통신을 사용하고 있다. 국내외에서 암호화 통신의 의무화 시행으로 인해 암호화 통신이 증가하는 추세가 나타나고 있다[2]. 이로 인해 내용이 숨겨져 있는 암호화 트래픽을 대상으로는 한계가 명확하고 보안에 민감한 데이터에 접근해야 하는 경우가 많아 사생활 침해의 위험이 있다는 한계가 있다. 머신러닝 기반 분류 방법은 트래픽의 크기, 도착 시간, 패킷 간 간격 등 다양한 특징을 학습하여 높은 정확도를 달성할 수 있었지만 대량의 학습 데이터가 필요하고, 학습 과정에서 높은 계산 자원을 요구하는 단점이 존재한다. 최근 연구로는 딥러닝 알고리즘을 사용한 응용 트래픽 분류 분야에서 높은 분류 정확도를 보이고 있다.

딥러닝 기반의 네트워크 트래픽 분류 방법은 복잡한 패턴을 자동으로 학습하고 높은 정확도를 제공하는 등 많은 장점을 가지고 있지만, 암호화된 데이터나 노이즈가 많은 환경에서 성능 저하를 겪는다는 단점이 있다. 암호화는 데이터의 구조적 무작위성을 증가시켜, 기존에 존재하던 패턴을 거의 완전히 감춘다. 딥러닝 모델은 패턴 인식을 통해 분류 작업을 수행하는데, 암호화된 데이터는 모델이 인식할 수 있는 명확한 패턴을 제공하지 않는다. 이러한 문제는 암호화로 인해 트래픽의 중요한 특징들이 숨겨지거나, 무작위성이 증가하기 때문에 모델이 유의미한 패턴을 효과적으로 학습하기 어렵게 만든다.

본 논문에서는 TLS 헤더 값을 기준으로 트래픽 데이터에서 암호화 적용 여부에 따라 데이터를 분리하고 암호화되지 않은 데이터만을 딥러닝 모델의 학습 데이터로 사용하는 방법을 제안한다. 이러한 접근은 무작위 값으로 이루어진 암호화 데이터로 인한 노이즈를 감소시키고 암호화되지 않은 데이터의 명확한 패턴 학습을 촉진하여 분류 성능을 향상시킬 수 있을 것으로 기대된다. 제안된 방법은 암호화되지 않은 데이터의 유용한 정보만을 활용함으로써 기존 딥러닝 기반 분류 방법의 한계를 극복하고, 다양한 네트워크 환경에서의 적용 가능성을 높이는 데 기여할 것이다.

1장의 서론에 이어 2장에서는 관련 연구에 대하여 설명한다. 3장에서는 암호화 여부에 따른 데이터 분리 방법에 대하여 설명한다. 4장에서는 실험 결과를 통하여 제안하는 방법론의 결론을 도출한다. 마지막으로 5장에서는 결론 및 향후 연구에 대하여 설명한다.

Ⅱ. 관련 연구

2.1 TLS

TLS(Transport Layer Security)는 인터넷 상에서 데이터 전송의 기밀성과 무결성을 보장하기 위해 널리 사용되는 암호화 프로토콜이다[3]. TLS는 HTTP, SMTP, FTP와 같은 애플리케이션 계층 프로토콜 위에서 동작하며, 데이터를 전송하기 전에 암호화하여 도청, 변조, 위변조를 방지한다. TLS는 여러 버전이 존재하며, 각 버전은 이전 버전의 보안 취약점을 개선하여 데이터 보안을 강화한다. TLS는 대칭 암호화, 비대칭 암호화, 해시 함수를 결합하여 데이터 보호를 수행하며, 이러한 암호화 메커니즘을 통해 네트워크를 통한 안전한 통신을 보장한다. TLS는 이러한 보안 기능을 구현하기 위해 클라이언트와 서버 간의 보안 연결을 설정하는 초기 단계로 TLS 핸드셰이크(TLS Handshake)를 수행한다.

TLS 핸드셰이크는 클라이언트와 서버 간의 보안 연결을 설정하기 위한 초기 단계로[4], 양측이 암호화 알고리즘, 세션 키, 인증서를 교환하고 합의하는 과정이며 그림1은 이를 표현한 그림이다. TLS 핸드셰이크는 주로 비대칭 암호화 방식을 사용하여 클라이언트와 서버 간의 인증을 수행하고, 이후의 데이터 전송에 사용할 대칭 키를 안전하게 생성한다. 핸드셰이크 과정은 클라이언트가 서버에 연결을 요청하는 “ClientHello” 메시지로 시작되며, 서버는 이에 응답하여 지원하는 암호화 알고리즘, 인증서, 세션 ID 등을 포함한 “ServerHello” 메시지를 전송한다. 이후 클라이언트는 서버의 인증서를 검증하고, 프리마스터 시크릿을 서버에 전송한 후, 양측은 대칭 키를 생성하여 보안 통신을 시작한다. TLS

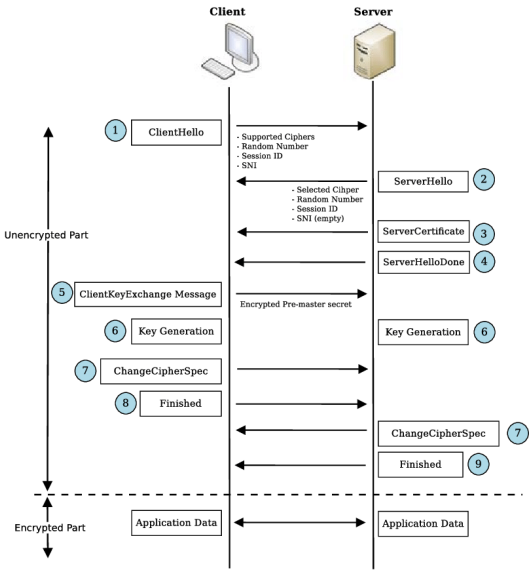

그림 1. TLS 핸드셰이크 과정
Fig. 1. Process of TLS Handshake

핸드셰이크는 보안 연결의 신뢰성과 무결성을 보장하는 핵심 절차로, 성공적인 핸드셰이크가 이루어진 후에야 암호화된 데이터 전송이 가능해진다.

2.2 딥러닝 기반 응용 트래픽 분류

본 절에서는 딥러닝 알고리즘을 사용한 응용 트래픽 분류 분야에서의 데이터셋 처리 방법을 간략히 소개한다.

1D-CNN(1-Dimensional Convolutional Neural Network)은 트래픽 데이터의 시퀀스 정보를 활용하여 패킷 또는 플로우 단위에서 트래픽을 분류하는 딥러닝 모델이다[5]. ISCX VPN 2016 데이터셋을 사용하여 트래픽을 분석하며, 플로우 또는 세션 단위로 데이터를 분리한 후, 각 단위의 시작부터 784 Byte까지의 데이터를 입력으로 사용한다. SAM(Sequence Attention Model)은 시퀀스 데이터를 다루는 트래픽 분류 모델로, 특히 패킷 수준의 입력을 통해 실시간 분류를 목표로 하는 모델이다[6]. WIDE, UNIBS, ISCX VPN 2016과 같은 데이터셋을 사용하여 패킷 단위로 데이터를 분리한 후, 각 패킷의 IP 헤더부터 50 Byte까지의 데이터를 사용한다. 추가로, IP 주소와 TCP/UDP 포트 정보는 마스킹하여 불필요한 정보는 제거한다. ET-BERT (Encrypted Traffic-BERT)는 BERT 모델의 변형으로, 암호화된 트래픽을 분류하는 데 특화된 모델이다[7]. ISCX VPN 2016을 포함한 7개의 다양한 데이터셋을 사용하며, Burst 단위로 패킷 또는 플로우를 분리한다. 전처리 과정에서 패킷 데이터는 2바이트 바이그램(Bi-gram)으로 이루어진 512개의 토큰으로 변환되며, Ethernet 헤더, IP 헤더, TCP 포트 등의 불필요한 정보는 제거한다.

언급한 연구들은 암호화된 트래픽 데이터셋을 대상으로 딥러닝 기법을 네트워크 트래픽 분류에 성공적으로 적용하였으나 비암호화된 데이터와 사실상 노이즈인 암호화된 데이터를 구분 없이 사용하였다. 암호화된 트래픽은 분류를 하는데 대표적인 특징이 부족하기 때문에 정상적인 학습을 방해할 수 있으며 명확한 패턴 인식을 방해할 수 있다[8]. 따라서, 암호화된 데이터와 비암호화된 데이터를 구분하여 분석하는 것이 중요하며 이에 대한 충분한 고려가 필요하다.

본 연구에서는 암호화 데이터 여부에 따른 트래픽의 특성 차이를 반영하여, 데이터셋에서 암호화되지 않은 데이터만을 추출하여 딥러닝 모델에 입력하는 새로운 방법론을 제안함으로써, 기존 연구의 한계를 보완하고자 한다.

Ⅲ. 본 론

3.1 방법론

TLS 헤더는 인터넷 상에서 안전한 데이터 전송을 보장하기 위해 설계된 필수적인 요소로 그 구조는 그림 2와 같다. 네트워크 통신의 기밀성과 무결성을 유지하는 데 중요한 역할을 하는 이 헤더는 메시지의 유형, 버전, 길이, 데이터 본문과 같은 여러 필드로 구성되어 있으며, 이러한 필드들은 데이터의 전송 상태와 보안 수준을 결정하는 데 사용된다. 특히, 메시지의 유형을 나타내는 ContentType 헤더 값은 해당 메시지가 암호화된 패킷인지 비암호화된 패킷인지를 구분하는 중요한 지표로 작용한다.

본 논문에서는 TLS 트래픽 분석을 위해 플로우에서 비암호화 헤더 값을 갖는 패킷만을 추출하는 방법론을 제안한다. 비암호화된 패킷은 change_cipher_spec(0x14)와 handshake(0x16)로, 암호화된 패킷은 alert(0x15)와

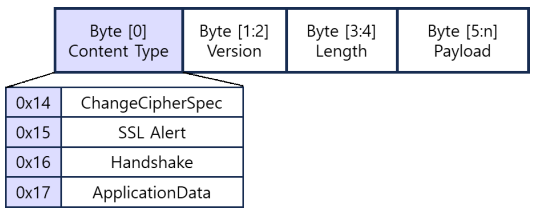

그림 2. TLS 헤더 구조
Fig. 2. Structure of TLS Header

application_data(0x17)로 ContentType 헤더 값에 따라 메시지의 유형이 구분된다. 이와 같은 구분을 통해 비암호화 패킷만을 선별하여 모델에 입력함으로써, 학습 과정에서 노이즈를 줄이고, 분류 성능을 향상시키는 것이 목표이다.

3.2 데이터셋

실험에 사용된 데이터셋은 ISCX VPN 2016으로 암호화된 응용 트래픽을 모아둔 공공 데이터셋이다[9]. 해당 데이터셋은 패킷 단위로 구성된 파일 형태이며, 16개의 응용 프로그램, 6개의 응용 카테고리, 2개의 VPN/NonVPN으로 구성되어 있다. 표 1은 데이터셋의 구성을 분류 업무에 따라 클래스를 나눈 것을 표로 나타낸 것이다. 수집된 트래픽은 세 가지 분류 업무 중에서 응용 프로그램 분류와 응용 카테고리에 대한 분류 연구를 수행한다.

표 1. ISCX 데이터셋 구성
Table 1. Information of ISCX VPN 2016 Datasets

Task	Class	#class
Encapsulation	VPN, non-VPN	2
Application	VoipBuster, FTP, Skype, BitTorrent, SFTP, Facebook,, SCP, Hangout, ICQ, Gmail, Email, AIM, YouTube, Netflix, Spotify, Vimeo	16
Category	VoIP, FileTransfer, Email, P2P, Streaming, Chat	6

3.3 전처리

공개 데이터셋 전처리 방법에 따라 패킷 단위의 pcap 파일을 입력으로 패킷 헤더의 5-tuples 정보가 같은 플로우 단위의 트래픽 파일로 변환한다[10]. 이후 본 논문에서 제안하는 방법론을 통해 세션 내에서 TLS 핸드셰이크 과정 이후의 암호화된 패킷들은 제거하는 과정을 거쳐, 최종적으로 10,163개의 플로우를 얻었다. 그리고 클래스를 특정할 때 강력한 특징이 될 수 있는 MAC 주소, IP 주소, TCP/UDP 포트번호 등을 마스킹하였다. 최종 데이터셋은 8:1:1의 비율로 train, validation, test 세트로 나누었으며, 표 2는 전처리 전후의 데이터셋 플로우 개수를 비교한 표이다.

표 2. 전처리 전후 플로우 비교
Table 2. Comparison of Flows According to Pre-processing

Method	# of Flows		
	TCP	UDP	Total
Raw-datasets	7,300	302,889	310,189
Proposed	5,086	5,077	10,163

3.4 분류 모델

실험에서 분류 모델은 트래픽 데이터를 자연어 단어처럼 표현하고 BERT 기반의 모델을 통해 해석하는 ET-BERT를 사용한다[7]. ET-BERT는 대규모 비라벨 데이터로부터 문맥적 트래픽 표현을 학습하며, 이후 소량의 라벨이 지정된 데이터로 미세 조정을 거쳐 다양한 트래픽 분류 작업을 수행한다. 12개의 트랜스포머 블록으로 구성되어 있으며, 각 블록은 12개의 어텐션 헤드를 포함하고 있다. 해당 모델은 공개 데이터셋 2종(ISCX VPN 2016[9], CIC-IDS2017[11])과 논문 저자들이 별도로 수집한 CSTNET-TLS 1.3 데이터셋을 대상으로 사전학습하였다.

Ⅳ. 실 험

본 장에서는 제안한 방법을 증명하기 위해 진행한 실험에 대하여 설명한다.

4.1 평가 지표

해당 실험에서는 결과의 평가와 비교를 위해 BalancedAccuracy(AC), Precision(PR), Recall(RC), F1을 포함한 4가지 평가 지표를 사용한다. 다음은 각 평가 지표를 계산하는 수식이다.

$$Balanced\ Accuracy = \frac{1}{2}\left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP}\right) \quad (1)$$

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

$$F1-score = 2 \times \frac{PR \times RC}{PR+RC} \quad (4)$$

4.2 실험 결과

표 3는 응용 프로그램 분류와 응용 카테고리 분류 2가지 업무에 대해 전처리를 하지 않은 데이터셋, 암호화/비암호화를 구분하지 않는 전처리를 거친 데이터셋, 제안하는 방법을 사용한 데이터셋을 대상으로 분류 실험을 진행한 결과이다. 두 분류 업무에서 제안하는 방법론을 사용한 쪽이 모두 더 높은 성능을 보였다.

그림 3은 응용 프로그램, 그림 4는 응용 카테고리

표 3. 분류 작업에 따른 결과 비교
Table 3. Comparison Results According to Classification Task

Task	Method	AC	PR	RC	F1
Category	Raw	0.7977	0.8213	0.7977	0.7956
	[10]	0.9178	0.9053	0.9178	0.9110
	Proposed	0.9796	0.9789	0.9796	0.9793
App	Raw	0.6950	0.7764	0.6950	0.7118
	[10]	0.8259	0.7694	0.7602	0.7626
	Proposed	0.9364	0.9527	0.9364	0.9435

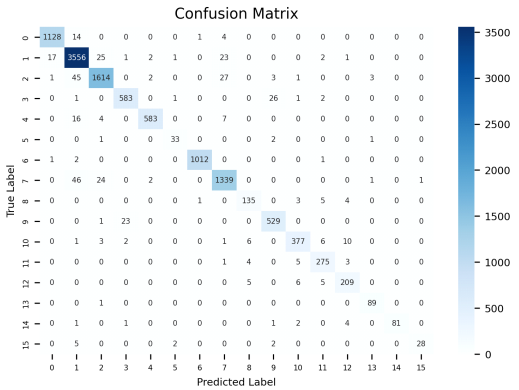

그림 3. 응용 프로그램 분류에 대한 혼동행렬
Fig. 3. Confusion Matrix of Application Classification

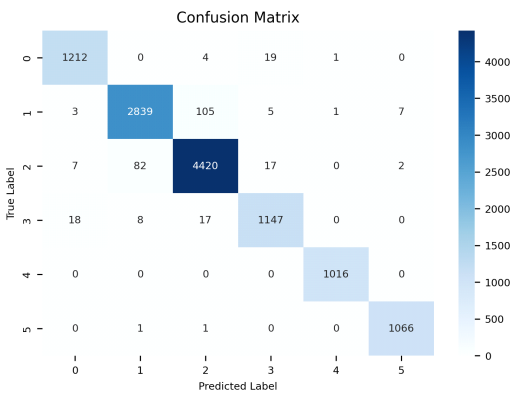

그림 4. 응용 카테고리 분류에 대한 혼동행렬
Fig. 4. Confusion Matrix of Category Classification

분류 결과에 대한 모델의 혼동행렬이다. 혼동행렬을 통해 모델의 성능을 분석한 결과, 대부분의 클래스에서 높은 True Positive 비율을 보이며 True Negative의 경우도 전반적으로 안정적인 성능을 유지하고 있어 모델이 대부분의 클래스에서 잘 예측하고 있음을 확인하였다. 그러나 일부 클래스 간의 혼동이 관찰되었고 이를 통해 모델이 특정 클래스 간의 경계를 명확하게 구분하지 못하였음을 짐작할 수 있다.

V. 결 론

본 논문은 딥러닝 기반의 네트워크 트래픽 분류 방법이 암호화된 데이터나 노이즈가 많은 환경에서 성능 저하를 겪는다는 문제를 해결하기 위해, TLS 헤더 값을 기준으로 트래픽 데이터에서 암호화 적용 여부에 따라 데이터를 분리하고 암호화되지 않은 데이터만을 딥러닝 모델의 학습 데이터로 사용하는 새로운 방법을 제안하였다. 실험 결과, 제안한 방법론을 적용했을 때 기본 전처리만을 사용한 결과보다 더 높은 정확도를 얻어내는 성과를 거두었다. 이는 본 연구에서 제안한 방법론이 일반적인 분류 방법보다 전체적인 정확도를 높일 수 있음을 보여주며, 딥러닝 기반 네트워크 트래픽 분류의 성능 향상에 기여할 수 있을 것이라 기대한다.

그러나, 본 연구는 TLS 필드 값을 기준으로 데이터를 구분하기 때문에 UDP 등 다른 프로토콜에 적용하기 어려운 제한이 있으며, SNI와 같은 비암호화 데이터가 클래스 특정을 용이하게 하여 과적합을 초래할 가능성도 존재한다. 따라서 향후 연구로 프로토콜에 상관없이 데이터 암호화 여부를 구분하고 이를 기준으로 비암호화 데이터를 추출하는 방법에 대하여 연구할 계획이다.

References

[1] M.-S. Lee, et al., "Deep learning-based traffic classification speed improvement through sequential data processing," *J. KICS*, vol. 47, no. 12, pp. 2096-2103, 2022. (https://doi.org/10.7840/kics.2022.47.12.2096)

[2] N. Ukeje, J. Gutierrez, and K. Petrova, "Information security and privacy challenges of cloud computing for government adoption: A systematic review," *Int. J. Inf Secur.*, vol. 23, no. 2, pp. 1459-1475, 2024. (https://doi.org/10.1007/s10207-023-00797-6)

[3] G. Apostolopoulos, V. Peris, and D. Saha, "Transport layer security: How much does it really cost?," *IEEE INFOCOM'99. Conf. Comput. Commun. Proc. Eighteenth Annual Joint Conf. IEEE Comput. and Commun. Soc.,*

The Future is Now (Cat. No. 99CH36320), vol. 2. IEEE, 1999. (https://doi.org/10.1109/INFCOM.1999.751458)

[4] T. Dierks and E. Rescorla, "*The transport layer security (TLS) protocol version 1.2*," No. RFC5246, 2008. (https://doi.org/10.17487/RFC5246)

[5] W. Wang, et al., "End-to-end encrypted traffic classification with one dimensional convolution neural networks," in *2017 IEEE Int. Conf. Intell. and Security Inf.*, pp. 43-48, 2017. (https://doi.org/10.1109/ISI.2017.8004872)

[6] G. Xie, Q. Li, and Y. Jiang, "Self-attentive deep learning method for online traffic classification and its interpretability," *Comput. Netw.*, vol. 196, 108267, 2021. (https://doi.org/10.1016/j.comnet.2021.108267)

[7] X. Lin, et al., "Et-bert: A contextualized datagram representation with pre-training transformers for encrypted traffic classification," in *Proc. ACM Web Conf.*, 2022. (https://doi.org/10.1145/3485447.3512217)

[8] S. Rezaei and X. Liu, "Deep learning for encrypted traffic classification: An overview," *IEEE Communi. Mag.*, vol. 57, no. 5, pp. 76-81, 2019. (https://doi.org/10.1109/MCOM.2019.1800819)

[9] G. Draper-Gil, A. H. Lashkari, M. S. I. Mamun, and A. A. Ghorbani, "Characterization of encrypted and VPN traffic using time-related features," in *Int. Conf. Inf. Syst. Secur. and Privacy*, pp. 407-414, 2016. (https://doi.org/10.5220/0005740704070414)

[10] U.-J. Baek, et al., "Preprocessing and analysis of an open dataset in application traffic classification," *2023 APNOMS IEEE*, pp. 227-230, 2023.

[11] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," *ICISSp*, vol. 1, pp. 108-116, 2018. (https://doi.org/10.5220/0006639801080116)

김 주 성 (Ju-Sung Kim)

2023년 : 고려대학교 컴퓨터융합소프트웨어학과 학사

2023년~현재 : 고려대학교 컴퓨터정보학과 석박사 통합 과정

<관심분야> 네트워크 관리 및 보안, 트래픽 모니터링 및 분석

[ORCID:0009-0002-4468-0717]

장 윤 성 (Yoon-Seong Jang)

2023년 : 고려대학교 컴퓨터융합소프트웨어학과 학사

2023년~현재 : 고려대학교 컴퓨터정보학과 석박사 통합 과정

<관심분야> 네트워크 관리 및 보안, 트래픽 모니터링 및 분석

[ORCID:0009-0006-7572-1541]

백 의 준 (Ui-Jun Baek)

2018년 : 고려대학교 컴퓨터정보학과 학사

2018년~현재 : 고려대학교 컴퓨터정보학과 석박사 통합 과정

<관심분야> 네트워크 관리 및 보안, 트래픽 모니터링 및 분석

[ORCID:0000-0002-4358-7839]

김 명 섭 (Myung-Sup Kim)

1998년 : 포항공과대학교 전자계산학과 학사
2000년 : 포항공과대학교 전자계산학과 석사
2004년 : 포항공과대학교 전자계산 학과 박사
2006년 : Dept. of ECS, Univof Toronto Canada
2006년~현재 : 고려대학교 컴퓨터정보학과 교수
<관심분야> 네트워크 관리 및 보안, 트래픽 모니터링 및 분석, 단일미디어 네트워크
[ORCID:0000-0002-3809-2057]