https://doi.org/10.33778/kcsa.2021.21.3.031

CNN을 활용한 Tor 네트워크 트래픽 분류

임 형 석*, 이 수 진**

요 약

Onion Router라고 알려진 Tor는 강한 익명성을 보장하기 때문에 각종 범죄행위뿐만 아니라 신속한 포트 검색 및 인증정보의 외부 유출 등 해킹 시도에도 활발하게 이용되고 있다. 따라서 범죄 시도를 조기에 차단하고 해킹으로부터 조직의 정보시스템을 안전하게 보호하기 위해서는 Tor 트래픽의 빠르고 정확한 탐지가 상당히 중요하다. 이에 본 논문에서는 CNN(Convolutional Neural Network)을 기반으로 Tor 트래픽을 탐지하고 트래픽의 유형을 분류하는 분류모델을 제안한다. 제안하는 분류모델의 성능 검증에는 UNB Tor 2016 데이터세트가 사용되었다. 실험을 진행한 결과, 제안하는 접근방법은 Tor 및 Non-Tor 트패픽을 탐지하는 이진분류에서는 99.98%, Tor 트래픽의 유형을 구분하는 다중분류에서는 97.27%의 정확도를 보여주었다.

Classification of Tor network traffic using CNN

Lim Hyeong Seok*, Lee Soo Jin**

ABSTRACT

Tor, known as Onion Router, guarantees strong anonymity. For this reason, Tor is actively used not only for criminal activities but also for hacking attempts such as rapid port scan and the ex-filtration of stolen credentials. Therefore, fast and accurate detection of Tor traffic is critical to prevent the crime attempts in advance and secure the organization's information system. This paper proposes a novel classification model that can detect Tor traffic and classify the traffic types based on CNN(Convolutional Neural Network). We use UNB Tor 2016 Dataset to evaluate the performance of our model. The experimental results show that the accuracy is 99.98% and 97.27% in binary classification and multiclass classification respectively.

Key words : Tor, CNN, Binary Classification, Multiclass classification
접수일(2021년 08월 27일), 수정일(2021년 09월 13일), 게재확정일(2021년 9월 28일)

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

1. 서 론

웹은 크게 표면웹(surface web), 딥웹(deep web) 및 다크웹(dark web)으로 구분된다[1]. 표면웹은 흔히 사용하는 구글이나 네이버 등 검색엔진으로 검색하여 접근할 수 있는 웹 페이지이며, 전체 웹의 4%에 불과하다. 나머지 96%는 딥웹 영역에 속하며, 일반 검색엔진으로는 접근하지 못한다[2]. 네트워크 감시 또는 트래픽 추적을 피할 목적으로 설계된 다크웹은 딥웹 영역에 속하지만, 암호화된 네트워크에 존재하며 별도 브라우저를 사용해야만 접속할 수 있다[3].

Tor는 다크웹에 접속하기 위해 사용되는 대표적인 브라우저로서, 다중 노드 간 트래픽을 암호화하여 목적지로부터 발신지의 추적을 어렵게 만든다[4]. 또한 강한 익명성을 보장하기 때문에 각종 범죄에도 활발하게 이용되고 있다[5]. Tor 상에서 운영되고 있는 5,200여개의 사이트를 무작위로 추출하였을 때 약 30%에 해당하는 1,547개의 사이트가 마약거래, 개인정보 수집, 청부살인, 아동포르노 유포, 불법 금융거래 등을 위해 운영되었고[6][7], 암호화 통신 특성을 이용하여 악성 소프트웨어 유포에도 이용되었다[8].

이러한 Tor 네트워크는 일반 네트워크를 우회하여 운영되기 때문에 범죄수사에 있어 어려움이 따른다. 특히 실제 트래픽에 대한 면밀한 분석을 통한 증거확보가 필요하나 암호화된 트래픽으로 인해 분석마저 쉽지 않다. 일반적으로 범죄 수사를 위한 디지털 증거확보는 ‘디지털 증거처리 표준가이드라인’에 의거해 하드디스크 내 네트워크 통신정보 분석을 통해 확보할 수 있다[9][10]. 그러나 Tor 네트워크 트래픽 분석은 전문화된 도구를 사용하더라도 많은 시간이 소요된다. 1TB 하드디스크를 기준으로 했을 때 디스크가 1개인 경우 약 11시간, 2개는 약 23시간, 3개는 약 41시간이 트래픽 분석을 위해 필요하다[10].

이러한 문제를 해결하기 위해 본 논문에서는 CNN(Convolutional Neural Network)을 활용하여 신속하게 Tor 트래픽을 탐지 및 분류함으로써 증거분석 및 범죄수사, 사이버보안 활동에 활용될 수 있는 모델을 제안하고자 한다. 본 논문의 구성은 다음과 같다. 2장에서는 Tor 네트워크에서 운영되는 웹사이트 식별과 트래픽 클래스 분류를 시도했던 기존 연구들을 정리하고, 3장에서는 트래픽 클래스 분류방안에 대해 설명한다. 4장에서는 제안된 방안을 적용하여 생성한 분류모델에 대한 성능 검증 결과를 분석하며, 마지막으로 5장에서 연구 결과를 요약하고 결론을 맺는다.

2. 관련 연구

Tor 트래픽 분류 분류에 관한 연구는 크게 Tor 네트워크에서 운영되는 웹사이트를 식별하는 연구와 트래픽의 유형을 분류하는 연구로 구분할 수 있다.

웹사이트 식별은 사용자와 Tor 네트워크 간 트래픽 정보를 수집한 후 트래픽에 포함된 TCP/IP 정보를 바탕으로 사용자가 방문 중인 웹사이트가 어떤 웹사이트인지를 구분한다. 특정 웹사이트와 관련된 특성 정보는 대부분 Alexa를 통해 수집하였다.

T. Wang 등[11]은 Tor 네트워크상에서 운영되는 상위 100개 사이트를 대상으로 SVM(Support Vector Machine) 알고리즘을 사용하여 91%의 성능을 달성하였다. V. Rimmer 등[12]은 상위 30개 사이트를 선정하고 다양한 알고리즘을 적용하여 웹사이트 식별을 시도하였다. 그 결과 SDAE(Stacked Denoising Auto Encoder)는 80%, CNN은 80%, LSTM(Long Short-Term Memory)은 76%, Cumulative 알고리즘은 78%의 식별 성능을 달성함을 확인하였다.

H. Oh 등[13]은 5개의 웹사이트 카테고리를 선정한 후 사용자와 중계 노드 사이에서 발생한 트래픽 정보 중 송수신 시간 간격, 패킷 길이, 버스트 등의 특징을 추출하여 학습에 활용하였다. 그 결과 결정트리(Decision Tree) 알고리즘은 90%, 랜덤 포레스트(Random Forest) 알고리즘은 92%, 엑스트라 트리(Extra Tree) 알고리즘은 93%의 정확도로 웹사이트를 식별함을 확인하였다.

Tor 트래픽의 유형을 분류하는 연구들은 주로 기계학습과 딥러닝을 활용하였다. A. Montieri 등[14]은 Anon-17 데이트세트를 대상으로 Naive Bayes 분류모델, C4.5, 랜덤 포레스트 알고리즘을 적용한 결과, Tor 트래픽과 Non-Tor 트랙픽을 분류하는 이진분류에서는 99.8%, 트래픽의 유형을 분류하는 다중분류에서는 73.9%의 성능을 달성하였다.

A. Lashkari 등[15]은 Skype, Facebook, Spotify,

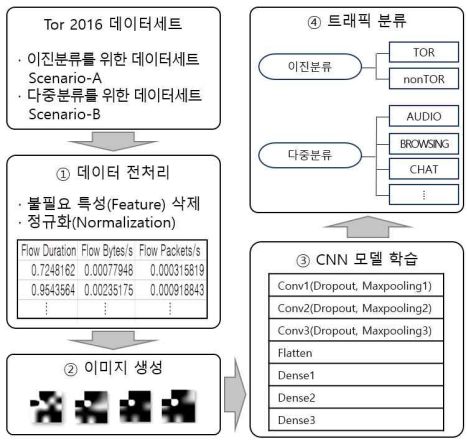
Gmail 등에서 19,497개의 Tor 웹서비스 데이터와 231,466개의 일반 웹서비스 패킷 데이터를 수집한 후, K-Nearest Neighbors 알고리즘, C4.5, ZeroR 알고리즘을 적용하여 이진분류 95%, 다중분류 75%의 성능을 달성하였다. A. Panchenko 등[16]은 775개의 다크웹사이트와 6,200개의 일반 웹의 데이터를 수집하고, SVM 알고리즘을 기반으로 다크웹 분류는 80%, 일반웹 분류는 73%의 성능을 달성하였다.

M. Kim 등[17]은 UNB(University of New Brunswick)에서 제공하는 Tor 2016 데이터세트[18]를 대상으로 이진분류 및 다중분류를 시도하였다. 먼저 pcap 파일을 108개의 특징을 가지는 21,085,710개의 데이터로 변환한 후 1차원 CNN모델에 학습시킨 결과 이진분류는 99.3%, 다중분류는 평균 93.72%의 성능을 달성하였다.

3. Tor 트래픽 분류방안

3.1 제안 절차

제안하는 Tor 트래픽 분류방안은 (그림 1)에서 보는 바와 같이 크게 4단계로 구성된다. 우선 UNB Tor 2016 데이터세트에 대해 분류에 영향을 미치지 않는 특성 제거 및 정규화 등의 전처리를 수행한다. 이어서 전처리된 데이터를 16비트 그레이 스케일 이미지로 변환하고, CNN 모델을 이용하여 학습을 실시한 후, 테스트 데이터를 대상으로 트래픽 분류를 수행한다.



(그림 1) 제안하는 Tor 트래픽 분류 절차

3.2 데이터세트

UNB Tor 2016 데이터세트는 이진분류 성능 평가를 위한 데이터세트인 'Scenario-A'와 다중분류 성능 평가를 위한 데이터세트인 'Scenario-B'로 구성되어 있다. Scenario-A에 포함된 nonTOR 데이터는 선행 연구에서 수집된 일반적인 트래픽만 포함하고 있다. Tor 트래픽은 총 5명의 사용자를 Tor 네트워크상의 노드로 구성한 후, AUDIO, BROWSING, CHAT, MAIL, P2P 등의 유형별 트래픽을 수집하였다. Scenario-A와 Scenario-B를 위한 데이터세트의 세부 구성은 <표 1>에서 보는 바와 같다.

<표 1> UNB Tor 2016 데이터세트 세부 구성

	Class	Number of data
Scenario-A	TOR	8,044
	nonTOR	59,790
	Total	67,834
Scenario-B	AUDIO	721
	BROWSING	1,604
	CHAT	323
	FILE-TRANSFER	864
	MAIL	282
	P2P	1,085
	VIDEO	874
	VOIP	2,291
	Total	8,044

3.3 데이터 전처리 및 이미지 생성

데이터 전처리는 Tor 2016 데이터세트에 포함된 csv 파일을 사용하였다. 우선 총 28개의 특성(feature) 중 IP(Internet Protocol), 포트(Port), 프로토콜(Protocol) 등 트래픽 분류에 영향을 주지 않는 5개의 특성(Source IP, Source Port, Destination IP, Destination Port, Protocol)을 제거하고 23개의 특성만을 대상으로 'min_max_scalar' 함수를 이용하여 정규화(normalization)를 수행하였다. 이어서 정규화된 값들을 행렬로 변환한 뒤 학습모델의 입력으로 사용하기 위한 28*28 크기의 16비트 그레이 스케일 이미지를 생성하였다.

3.4 신경망

본 논문에서 제안하는 분류모델 생성을 위해서는 딥러닝의 대표적인 알고리즘으로 별도의 특성 분석(feature engineering) 과정 없이도 높은 분류 성능을 보장하는 CNN을 활용하였다.

반복 실험을 통해 최적화된 신경망의 세부 구성은 <표 2>에서 보는 바와 같다. 28*28 사이즈의 이미지를 입력으로 받으며, 3개의 컨볼루션 계층(convolutional layer)과 이미지 특성 맵의 차원 축소를 위한 크기 2의 맥스풀링 계층(maxpooling layer)을 각 컨볼루션 계층 뒤에 배치하였다.

완전연결계층(fully-connected layer)에서는 활성화 함수로 ReLU[19]를 사용하고, 출력계층은 Softmax를 사용하였다. 신경망의 최적화 알고리즘은 Adam[20]을 사용하고, 학습률은 Adam의 기본값인 0.0005를 적용하였다.

<표 2> 제안 모델의 신경망 세부 구성

Layer	Output shape	Paramater
Conv1	(None, 64, 26, 26)	640
Dropout	(None, 64, 26, 26)	0
Maxpooling1	(None, 64, 13, 13)	0
Conv2	(None, 128, 11, 11)	73,856
Dropout	(None, 128, 11, 11)	0
Maxpooling2	(None, 128, 5, 5)	0
Conv3	(None, 256, 3, 3)	295,168
Dropout	(None, 256, 3, 3)	0
Maxpooling3	(None, 256, 1, 1)	0
Flatten	(None, 256)	0
Dense1	(None, 64)	16,448
Dense2	(None, 32)	2,080
Dense3	(None, 8)	264
Total parameter		388,456
Trainable parameter		388,456
Non-trainable parameter		0

4. 실험 및 평가

이진분류를 위한 Scenario-A, 다중분류를 위한 Scenario-B 데이터세트의 모든 데이터를 제안된 모델 신경망에 각각 학습시켜 이진분류와 다중분류 성능을 확인하였다.

4.1 실험환경

실험은 Windows 10 Home 64bit 운영체제, Intel(R) Core(TM) i7-8750H CPU, 16G RAM 사양의 Laptop에서 Google Colaboratory를 이용한 GPU 하드웨어 가속기를 기반으로 진행되었다.

실험 진행을 위한 데이터는 전체 데이터 샘플을 Pycham에서 제공하는 랜덤 추출함수 Ramdom state(값 0)를 이용하여 일정 비율로 나누고 무작위로 추출하여 구성하였다. 학습(Train)과 테스트(Test)에 사용하는 데이터는 85:15의 비율로 구분하였으며, 학습용 데이터는 다시 학습과 검증(Validation)용 데이터를 80:20의 비율로 나누었다. 실험 데이터 샘플의 세부 구성은 <표 3>과 <표 4>에서 보는 바와 같다.

<표 3> Scenario-A를 위한 실험데이터 구성

Class	Train	Test	Validation	Total
TOR	5,469	1,207	1,368	8,044
nonTOR	40,656	8,969	10,165	59,790
Total	46,125	10,176	11,533	67,834

<표 4> Scenario-B를 위한 실험데이터 구성

Class	Train	Test	Validation	Total
AUDIO	489	109	123	721
BROWSING	1,090	241	273	1,604
CHAT	219	49	55	323
FILE-TRANSFER	587	130	147	864
MAIL	191	43	48	282
P2P	737	163	185	1,085
VIDEO	593	132	149	874
VOIP	1,557	344	390	2,291
Total	5,463	1,211	1,370	8,044

4.2 성능평가 지표

분류 성능을 평가하기 위한 지표로는 'Precision', 'Recall', 'F1-score' 및 'Accuracy'를 사용하였다. Precision은 '정밀도'를 의미하며, 분류모델이 True로 분류한 것 중 실제 True인 것의 비율로서 다음과 같이 산출된다.

$$Precision = (TP)/(TP+FP) \quad (1)$$

Recall은 '재현율'이라고도 하며, 실제 True인 것 중 모델이 True라고 예측한 것의 비율이다. 식으로 표현하면 다음과 같다.

$$Recall = (TP)/(TP+FN) \quad (2)$$

F1-score는 성능 평가를 위해 사용되는 2가지의 지표 Precision과 Recall의 조화평균을 의미하는 지표로서, 다음과 같이 표현된다.

$$F1-score = 2 \times \frac{1}{\frac{1}{Precision}+\frac{1}{Recall}} = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)} \quad (3)$$

Accuracy는 '정확도'를 의미한다. True를 True로 예측한 경우와 False를 False라고 정확히 예측한 경우도 포함하는 지표로서, 다음과 같이 표현된다.

$$Accuracy = (TP+TN)/(TP+FN+FP+TN) \quad (4)$$

4.3 실험결과 및 분석

'TOR' 및 'nonTOR' 클래스를 분류하는 이진분류와 8개의 트래픽 유형을 분류하는 다중분류 실험을 진행하였다. 전처리 후 이미지화한 파일을 CNN 분류모델 입력으로 사용하여 학습을 진행하였으며, 동일 데이터 전체를 이용하여 20회 반복 실험을 진행하고 산출된 값의 평균값을 적용하였다. Scenario-A와 Scenario-B 데이터세트에 대한 분류 성능 평가 결과는 <표 5>과 <표 6>에서 보는 바와 같다.

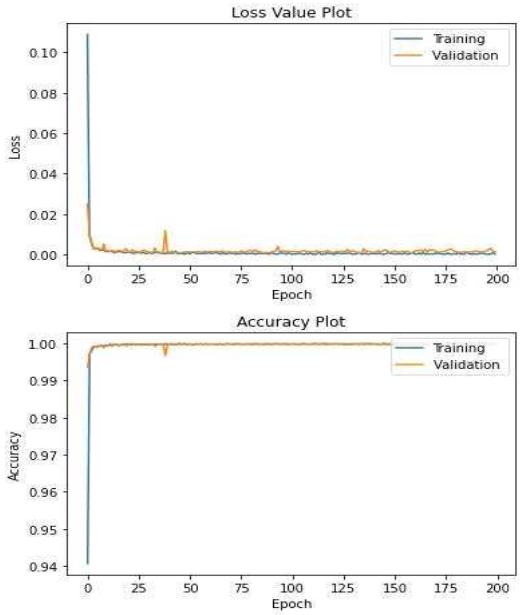
<표 5> Scenario-A에 대한 성능 평가 결과

Class	Precision	Recall	F1-score
TOR	99.92%	99.92%	99.92%
nonTOR	99.99%	99.99%	99.99%
Average	99.95%	99.95%	99.95%
Accuracy		99.98%	

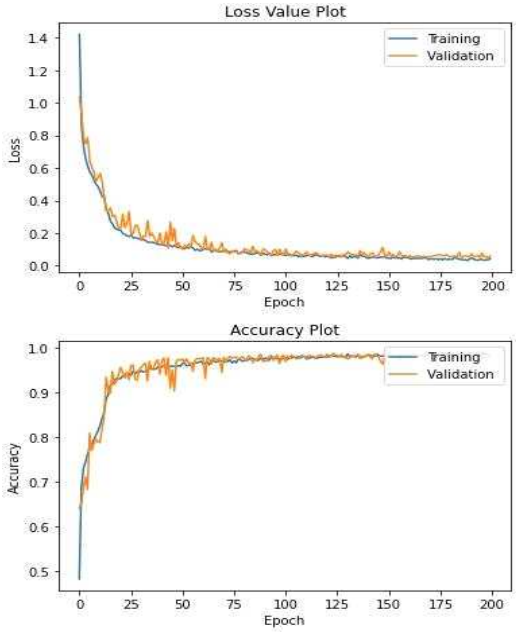
<표 6> Scenario-B에 대한 성능 평가 결과

Class	Precision	Recall	F1-score
AUDIO	95.61%	100%	97.76%
BROWSING	97.91%	97.1%	97.5%
CHAT	82.35%	85.71%	84%
FILE-TRANSFER	100%	96.92%	98.44%
MAIL	86.96%	93.02%	89.89%
P2P	99.38%	97.55%	98.45%
VIDEO	96.21%	96.21%	96.21%
VOIP	99.42%	99.13%	99.27%
Average	94.73%	95.7%	95.19%
Accuracy		97.27%	

학습 간 200 Epoch 이후부터 과적합이 발생하여 Epoch는 200으로 설정하고 학습을 진행하였다. 학습 Epoch 진행에 따른 손실률(Loss)과 정확도(Accuracy) 변화 추이는 (그림 2)와 (그림 3)에서 보는 바와 같다. Scenario-A와 Scenario-B 데이터세트 모두 학습과 Validation의 손실률 및 정확도가 유사하게 나타나는 것을 확인할 수 있다.



(그림 2) Scenario-A의 학습 손실률 및 정확도



(그림 3) Scenario-B의 학습 손실률 및 정확도

<표 6>에서 확인할 수 있는 바와 같이, 'CHAT' 및 'MAIL' 클래스의 Precision 지표가 다소 낮게 나타났는데, 이는 두 클래스의 데이터 수가 다른 클래스의 데이터 수에 비해 상대적으로 적었기 때문으로 판단된다.

이진분류의 오차행렬(Confusion Matrix)는 <표 7>에서 보는 바와 같고, 대부분의 트래픽이 실제 클래스로 정확하게 분류되었음을 확인할 수 있다. <표 8>는 다중분류 오차행렬을 보여주고 있으며, 대부분의 클래스에서 일부 트래픽이 잘못 분류된 것을 확인할 수 있다. 특히 'CHAT', 'VIDEO' 및 'BROWSING' 클래스에서 잘못 분류된 트래픽이 다수 확인되었다.

<표 7> 이진분류 오차행렬

True class	Predicted class	
	TOR	nonTOR
TOR	1,206 99.92%	1
nonTOR	1	8,968 99.99%

<표 8> 다중분류 오차행렬

True class	Predicted class							
	CHAT	AUDIO	VIDEO	FILE-TRANSFER	P2P	MAIL	BROWSING	VOIP
CHAT	42 82.3%	0	3	0	0	0	3	1
AUDIO	0	109 95.6%	0	0	0	0	0	0
VIDEO	2	0	127 96.2%	0	0	2	0	1
FILE-TRANSFER	1	0	0	126 100%	0	3	0	0
P2P	1	1	0	0	159 99.3%	1	1	0
MAIL	1	0	0	0	1	40 86.9%	1	0
BROWSING	3	4	0	0	0	0	234 97.9%	0
VOIP	1	0	2	0	0	0	0	341 99.4%

5. 결 론

본 논문에서는 각종 범죄 및 해킹행위에 악용되는 Tor 네트워크 트래픽을 신속하고 정확하게 식별하기 위해 네트워크에서 수집된 데이터만을 이용하여 CNN 기반으로 탐지 및 분류하는 기법을 제안하였다. 제안하는 기법은 기존에는 포렌식 전문가가 각종 기법과 도구를 사용하여 수작업으로 분석했던 부분을 딥러닝 기술을 기반으로 신속하고 정확하게 분류할 수 있도록 개선하였다는 점에서 향후 Tor 네트워크를 이용하는 범죄수사에 큰 도움이 될 것으로 판단된다.

제안된 기법의 효율성을 검증하기 위해 UNB Tor 2016 데이터세트를 정규화하고 16비트 그레이 스케일 이미지로 변환한 후 CNN 모델에 학습시켜 트래픽 분류 성능을 확인한 결과 이진분류는 99.98%, 다중분류는 97.27%의 성능이 보임을 확인하였다. 이러한 결과는 동일한 데이터세트로 진행했던 기존연구[17]에 비해 이진분류는 0.68%P, 다중분류는 3.55%P의 성능 향상을 달성한 것이다.

향후 연구에서는 'CHAT' 및 'MAIL' 등 일부 클래스의 데이터 수가 다른 클래스 대비 적어 분류 성능 또한 낮게 나타났던 문제점을 개선하기 위해 GAN (Generative Adversarial Network)을 기반으로 희소 데이터의 개수를 늘려 충분한 학습이 이루어지도록 함으로써 전체적인 분류 성능을 향상시킬 것이다.

참고문헌

[1] A. Gupta, S. B. Maynard, and A. Ahmad, “The Dark Web Phenomenon: A Review and Research Agenda”, ACIS 2019 Proceedings, 1, 2019.

[2] Kristin Finklea, “Dark Web”, U. S. Congressional Research Service Report, 2017. 3.

[3] S. Kaur and S. Randhawa, “Dark Web: A Web of Crimes”, Wireless Personal Communications, Vol. 112, 2020.

[4] K. Rathod, and H. Suthar, “Traffic Analysis and Relay Finding in Tor Survey”, Multidisciplinary International Research Journal of Gujarat Technological University, Vol. 2, No. 1, pp. 34-43, 2020.

[5] L. Basyoni, N. Fetais, A. Erbad, A. Mohamed, and M. Guizani, “Traffic analysis attacks on Tor: a survey”, 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), pp. 183-188, 2020

[6] AhnLab, “ASEC REPORT VOL. 50”, https:// www.ahnlab .com/kr/site/securityinfo/asec/asecReportList.do, 검색일: 2020. 8. 5, pp.1-25, 2014.

[7] D. Moore, and T. Rid, “Cryptopolitik and the Darknet”, Survival, Vol.58, no.1, pp.20-25, 2016.

[8] Z. Cao, G. Xiong, Y. Zhao, Z. Li, and l. Guo, “A Survey on Encrypted Traffic Classification”, International Conference on Applications and Techniques in Information Security, pp. 73-81, 2014.

[9] Y. Shin, and S. Shin, “An Empirical Study on Massive Forensic Services”, Internet and Information Security, Vol.1, No.4, pp. 83-100, 2010.

[10] M. Kim, “Limitations and Improvements of Adoption Criteria for Digital Forensic Evidence”, Convergence Security Journal, Vol.18, No.4, pp. 36-43, 2018.

[11] T. Wang, and I. Goldberg, “Improved website finger-printing on tor”, Proceedings of the 12th ACM workshop on Workshop on privacy in the electronic society, pp. 201-202, 2013.

[12] V. Rimmer, D. preuveneers, M. Juarez, T. Van Goethem, and W. Joosen, “Automated Website Fingerprinting through Deep Learning”, arXiv preprint arXiv:1708.06376, pp. 1-15, 2017.

[13] H. Oh, D. Hwang, and W. Kim, “Traffic Sequence Vectorization and Ensemble Algorithm Classification for Tor Website Fingerprinting”, Journal of The Institute of Electronics and Information Engineers Vol. 57, No. 5, pp. 59-61, 2020.

[14] A. Montieri, D. Ciuonzo, G. Aceto, and A. Pescape, “Anonymity Services Tor, I2P, JonDonym: Classifying the Dark (Web)”, IEEE Transactions on Dependable and Secure Computing, Vol.17, No.3, pp. 1-14, 2018.

[15] A. Lashkari, H. Draper-Gil, M. S. I. Mamun, and A. Ali, “Characterization of Tor Traffic using Time based Features", International Conference on Information Systems, Security and Privacy(ICISSp), pp. 253-263, 2017.

[16] A. Panchenko, F. Lanze, J. Pennekamp, T. Engel, A. Zinnen, M. Henze, and K. Wehrle, “Website Fingerprinting at Internet Scale”, Network and Distributed System Security Symposium(NDSS), pp. 1-15, 2016.

[17] M. Kim, and A. Anpalagan, “Tor Traffic Classification from Raw Packet Header using Convolutional Neural Network”, 2018 1st IEEE International Conference on Knowledge Innovation and Invention(ICKII), pp. 187-190, 2018.

[18] University of New Brunswick, “Tor-nonTor dataset (ISCXTor2016)”, https://www.unb.ca/cic/datasets/andmal 2017.html, 2016.(검색일 : 2020. 7. 5)

[19] V. Nair, and G. E. Hinton, “Rectified Linear Units Improve Restricted Boltzmann Machines”, International Conference on Machine Learning (ICML), pp. 807-814, 2010.

[20] Kingma, P. Diederik, and Jimmy Ba, “Adam: A Method For Stochastic Optimization“, arXiv preprint arXiv:1412.6980, pp. 1-15, 2015.

[저자소개]

임 형 석 (Hyeong-seok Lim)
2015년 2월 강원대학교
컴퓨터과학과 학사
2020년 3월~현재 국방대학교
컴퓨터공학전공 석사과정
email : dla9252@gmail.com

이 수 진 (Soo-jin Lee)
1992년 3월 육군사관학교
전산학과 학사
1996년 2월 연세대학교
컴퓨터과학과 석사
2006년 2월 한국과학기술원
전산학과 박사
2006년~현재 국방대학교
국방과학학과 교수
email : cyberkma@gmail.com