Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.29, NO.4, Aug. 2019 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2019.29.4.785

네트워크 데이터 정형화 기법을 통한 데이터 특성 기반 기계학습 모델 성능평가

이 우 호,[1†] 노 봉 남,[1] 정 기 문[2‡]
[1]전남대학교 정보보안협동과정, [2]한국과학기술정보연구원

Performance Evaluation of a Machine Learning Model Based on Data Feature Using Network Data Normalization Technique

Wooho Lee,[1†] BongNam Noh,[1] Kimoon Jeong[2‡]
[1]Interdisciplinary Program of Information Security, Chonnam National University
[2]Korea Institute of Science and Technology Information

요 약

최근 4차 산업 혁명 기술 중 하나인 딥러닝(Deep Learning) 기술은 보안 분야에서는 탐지하기 어려운 네트워크 데이터의 숨겨진 의미를 식별하고 공격을 예측하는 데 사용되고 있다. 침입탐지에 사용될 딥러닝 알고리즘을 선택하기 전에 데이터의 속성과 품질 분석이 필요하다. 학습에 사용되는 데이터의 오염여부에 따라 탐지 방법에 영향을 주기 때문이다. 따라서 데이터의 특징을 파악하고 특성을 선정해야 한다. 본 논문에서는 네트워크 데이터 셋을 이용하여 악성코드의 단계적 특징을 분석하고 특성을 추출하여 딥러닝 모델을 적용하였을 때 각 특성이 성능에 미치는 영향을 분석하였다. 네트워크 특징에 따른 특성들의 비교에 대한 트래픽 분류 실험을 진행하였으며 선정한 특성을 기반으로 96.52% 정확도를 분류하였다.

ABSTRACT

Recently Deep Learning technology, one of the fourth industrial revolution technologies, is used to identify the hidden meaning of network data that is difficult to detect in the security arena and to predict attacks. Property and quality analysis of data sources are required before selecting the deep learning algorithm to be used for intrusion detection. This is because it affects the detection method depending on the contamination of the data used for learning. Therefore, the characteristics of the data should be identified and the characteristics selected. In this paper, the characteristics of malware were analyzed using network data set and the effect of each feature on performance was analyzed when the deep learning model was applied. The traffic classification experiment was conducted on the comparison of characteristics according to network characteristics and 96.52% accuracy was classified based on the selected characteristics.

Keywords: IDS, Deep learning, Data normalize

I. 서 론

최근 사이버 위협은 특정 기기를 대상으로 보다 정교하게 준비하여 이루어지고 있으며, PC뿐만 아니라 사물인터넷 등의 다양한 시스템에 대한 위협이 현실화되고 있다. 사물인터넷은 스마트 홈, 자동차, 항공기, 무기 체계 등에 접목되면서 생활 속 사용자의 편의성과 효용성이 증가하고 있다. 하지만 최근에

Received(04. 04. 2019), Modified(06. 17. 2019), Accepted(06. 17. 2019)

† 주저자, leeouho@naver.com
‡ 교신저자, kmjeong@kisti.re.kr(Corresponding author)

는 사이버 위협 또한 증가하고 있다.[1]

2014년 이후로 봇넷을 이용한 공격이 급격히 증가하고 있다. 허니팟을 이용하여 텔넷과 웹 컨텐츠를 살펴본 결과, 공격의 대부분이 실제로 사물인터넷 기기를 대상으로 공격된 것으로 나타났다[3]. 하지만 기존의 네트워크 침입 탐지 시스템 기술의 발전에도 불구하고, 현재의 솔루션은 다양한 환경에서 발생하는 공격을 완벽하게 탐지하기에는 기술이 부족하다[4].본 연구에서는 다양한 트래픽 특징을 분석하여 특성을 선정하고 트래픽 분류 성능을 분석한다. 세션으로 트래픽을 탐지할 경우 다음과 같은 장점이 존재하며, 이를 통해 여러 행위에 대한 흐름을 걸러낼 수 있다[21].

1. 합법적인 호스트와 통신을 제외한 비정상 트래픽을 걸러 낼 수 있다. 현재의 연결된 상태의 네트워크 파악할 수 있다.
2. 긴 TCP 세션에서 짧은 TCP 세션(패킷 4개 이하)로 분리하거나 PSH 플래그처럼 세션이 합법적이라는 신호를 찾을 수 있다.
3. 명령어 파일 전송 등으로 트래픽을 세분화를 하면 class의 특징을 파악할 수 있으며 특성을 추출하는데 효율적이다. 또한 분류 정확성을 높이기 위해 알고리즘을 미세 조정하여 선별할 수 있는 장점이 있으며, 필터링 작업을 확대해서 트래픽분류에 효율성을 높일 수 있다.
4. 단순히 양에 대한 임계치를 사용하여 처리해야할 트래픽 양과 잡음을 줄일 수 있다.

최근 머신러닝을 이용하여 트래픽량이 적은 데이터셋에서 공격 트래픽을 분류하기 위한 연구가 활발히 진행되고 있다. 이러한 연구들은 네트워크에 접근하는 IP 주소나 URL 수와 같은 통계 값을 기반으로 구별하는 방법을 적용하여 분류하고 있다.

본 논문에서는 네트워크 특징에 따른 특성을 선정하고 선정된 특성이 트래픽 분류에 미치는 영향에 대해서 서술하였다. 논문의 구성은 2장에서는 최근에 연구되어지고 있는 딥러닝을 이용한 트래픽 분류에 대한 연구에 대해 서술하고 3장에서는 특성 추출 및 분석 4장에서는 실험에 대한 내용에 대해 서술하고 마지막으로 5장에서는 결론을 맺는다.

II. 관련연구

W.Haider는[5]에서 DFR(Data Feature Retrieva)와 KNN을 이용한 기계학습 알고리즘을 이용한 HADS(Host-based anomaly detection systems)을 제안하였다. 본 시스템은 Xie의 통계적 방법을 이용한 탐지와 비교하였을 때 데이터 특성 추출을 위한 계산 비용을 최소화하여 계산에 소요되는 시간을 감소시키는 결과를 보였다.[6] 하지만 탐지율은 12%~2% 낮아지고 오탐율은 5% 높아지는 한계점을 보인다.

M.Guerroumi는[7]에서 무선 네트워크 환경에서 싱크홀 공격에 대응하기 위하여 클러스터 기반 라우팅 프로토콜을 보호하기 위한 방법을 제안하였다. IDS에서 공격을 탐지하기 위하여 중요 특성을 추출하고 N.A-Detection 알고리즘을 이용한 네트워크 트래픽 부하에 영향을 받지 않는 탐지 방법을 제안하였다.

Yuxuan Luo[8]는 실제 시나리오에서는 HTTP 트래픽 데이터 집합이 불명확하여 탐지가 어렵다. 학습된 학습 모델과 HTTP 페이로드 데이터를 기반으로 하는 PU 학습 (Positive 및 Unlabeled 학습)을 결합한 새로운 웹 이상 탐지 방법을 제안하였다.

WANG는[9]에서 세션 트래픽 선두 수백 바이트만을 사용하여 CNN 모델을 적용하는 방법을 제안하였다. 이는 초기 단계의 악성코드에 의해 발생되는 트래픽을 탐지하는 기능을 가진다. CNN알고리즘을 이용하여 이미지 분류 방법을 사용하기 때문에 프로토콜에 독립적이라는 특징이 있다.

Li는[10]에서 기존 악성코드 탐지에 특징 추출을 적절하게 적용하지 못하여 탐지율과 정확도가 낮았던 문제를 해결하기 위하여 AE와 DBN을 결합한 하이브리드 악성 코드 검출모델을 제안하였다. AE를 차원 감소에 사용하고 RBM을 통한 자동 측정 방법에 사용하였다. KDD-Cup-99 데이터셋을 사용하여 실험한 결과 AE를 사용하여 데이터의 크기를 축소하는 것이 효과적이었으며 탐지 정확도가 향상 될 수 있음을 보였다.

Tao는[11]에서 대용량 네트워크 트래픽 분류에 DAE알고리즘을 적용하여 데이터 차원 축소방법을 제안하였다. 또한 BPNN, SVM과 같은 분류 알고리즘의 정확도 향상을 비교 실험하여 효율성을 향상시키는 것을 연구하였다.

Tang는[12]에서 NSL-KDD 데이터셋을 사용하

여 SDN 환경에서 IDS를 적용하기 위해 DNN 방법을 적용하였다. 41개의 특성 중 duration, protocol_type, src_bytes, dst_bytes, count 및 src_count 등 6가지 기능을 선정하여 실험한 결과 손실량은 7.4%이고 정확도는 91.7%로 41개 전체 특성을 이용하여 분류하는 것보다 더 나은 성능을 보였다.

W.jung는[13]에서는 zero-day flash 악성코드를 탐지하기 위한 방법으로 악성코드 데이터로부터 특징을 추출하고 RNN 기법을 적용한 모델을 제안하였다.

L.Nataraj는[14]에서는 KNN을 이용한 악성코드 분류 방법을 제안하였다. 악성코드 파일을 이미지화 하고 KNN알고리즘을 통해 악성코드의 유형을 분류 할 수 있지만 기존의 머신러닝에 적용시키기 위하여 이미지의 특징을 추가적으로 추출해야 한다는 단점이 있다.

KIM JI는[15] KDD Dataset1999에서 GRU를 이용한 침입탐지 방법 연구는 hidden layer 별 분석을 통해 높은 영역에 대한 알고리즘(RNN, LSTM, GRU)를 대상으로 하여 비교 실험 하였다. KDD dataset을 이용하여 DoS, R2L, Normal 등에 대하여 특성을 추출하고 군집화 하였으며 hyperparameter를 이용하여 각 수치들에 대한 관계를 분석한 논문이다.

KIM JI는[16] LSTM의 성능 향상을 위해 KDD Dataset1999를 이용하였고 RNN과 결합한 LSTM-RNN 알고리즘기반 IDS를 제안하였다.

특성추출을 위하여 상관관계를 이용한 특성을 생성하였고 평가를 위해 다른 알고리즘(CRNN, RNN, RBNN, KNN, SVM, Basian)과 비교 실험 하였다.

BEDIAKO는[17] LSTM을 이용한 DDoS 공격에 대한 탐지 방법을 제안했다. UNB ISCS Intrusion Detection Evaluation 2012 dataset을 이용하여 DDoS 공격을 집중적으로 연구하였으며 20개의 특성을 추출하여 DDoS 탐지 및 추출 기준을 서술하였다.

Liu는 CNN 기반의 침입 탐지 알고리즘을 개발하여 IDS 모델을 평가했다. 학습 단계에서는 KDD Cup 1999 데이터 세트에서 추출된 특징을 이용해서 데이터 세트를 생성하고 보다 편리한 CNN 학습을 위해 테스트 데이터에 대해 2차원 처리를 수행했다.CNN을 이용한 분류 방법 중 가장 높은 97.7%의 탐지율을 증명했다[18].

기존의 연구에서는 지도학습을 기반의 비정상 트래픽 분류의 한계점인 학습되지 않은 공격 또는 변형된 공격탐지를 해결하기 위한 연구가 진행되고 있다. 본 연구에서는 트래픽의 특징을 분석하고 특성을 추출하여 공격을 탐지하기 위한 방법으로 다양한 특성 중 6가지 공통적인 특성은 SRC_Port, DST_Port, DIR, Session Winsize, Timestamp을 선정하였으며 3가지 시나리오에 따른 분류를 Confusion Matrix와 F1-score를 이용하여 평가하였다.

III. Feature Selection

3.1 분석과정

본 연구에서 네트워크 특징을 분석하고 특성을 선정하여 각 시나리오 별로 분류를 진행한다. Fig. 1은 트래픽 분류 및 분석을 위한 데이터 전처리/정형화 과정이다. 데이터 패밀리 식별은 딥러닝 기반 모델을 이용하는데 분류과정은 다음과 같다.

1. Collector Server는 기기로부터 서버로 하드웨어 정보를 수집하며, 동시에 네트워크 패킷을 수집한다.
2. 수집된 정보는 특성 추출 필터를 이용하여 통합 작업을 수행한다. Training과 Test 데이터는 6:3 비율로 나눈다.
3. 수집된 Training dataset은 데이터 전처리 과정을 통해 TF-IDF 벡터형식으로 바꿔준 후 array를 생성하여,n-gram을 이용하여 float32로 변환하고 ScIkit-learn의 minmax함수를 이용하여 0~1사이의 값으로 변환하여 준다.
4. 결합된 데이터 셋은 정규화 과정을 거쳐 csv파일 형태로 생성된다.
5. CSV파일은 상관관계 분석을 통해 데이터 분포를 정규분포로 변경하여 준다.
6. 정규화된 데이터는 CNN 모델을 이용하여 학습하게 되며 test data로 이용하여 검증하는 단계를 거친다.
7. 생성된 모델들은 저장된 특징을 바탕으로 시나리오 별 패밀리 분류를 실시한다.
8. 패밀리 구분 모델들은 도출된 결과가 일치하지 않을 경우 다시 2개 이상의 모델들의 동일한 탐지 값을 결과로 합의한다.

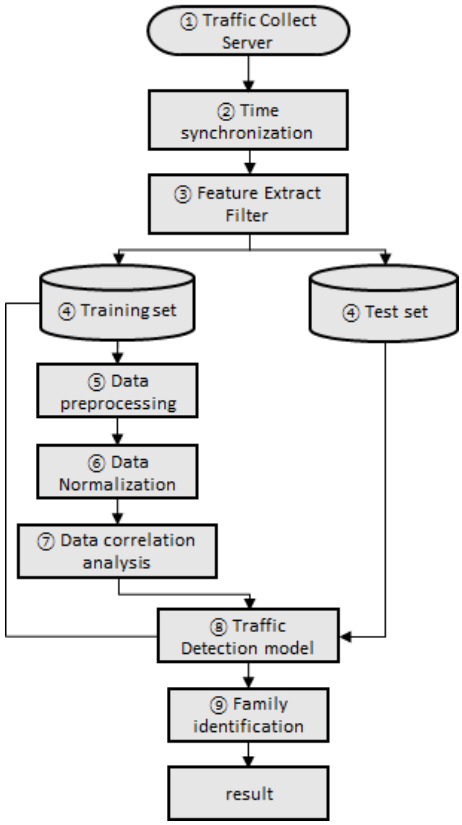

Fig. 1. Traffic Collection and Data Analysis Process

3.2 네트워크 특성 추출 과정

네트워크 특징의 경우 패킷 수집 도구를 이용하여 정적분석을 통해 IP, port, session 특성을 선정하였다. 트래픽 수집 도구를 이용하여 약 40000개의 패킷에 대한 scan, loaded, C&C 명령어에 따른 트래픽 변화 등과 같은 특징을 추출하였다.

Fig. 2는 트래픽 수집 도구의 전처리과정이다. 패킷 수집도구에서는 패킷을 수집하고 pcap파일 형태로 데이터를 수집한다. 수집된 데이터는 필터를 이용하여 세션을 검사하고 Src IP, Dst IP, 포트번호, 프로토콜, 세션 데이터 등의 특성을 기반으로 Table 형태로 수집하게된다.

수집된 데이터는 범주형 데이터로 구성되어있어 TF-IDF를 이용하여 Vector로 변환시킨 후 배열로 변환한다. 변환된 데이터는 데이터 정형화과정을 통해 raw 데이터로 입력된다.

본 연구에서는 다양한 연구에서 공통적으로 사용

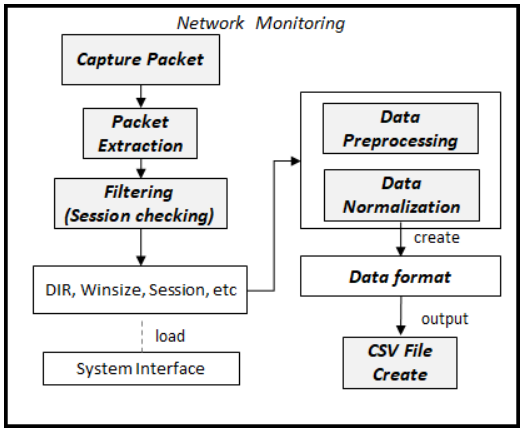

Fig. 2. Traffic Collection and Preprocessing

되는 패킷 데이터의 여러 특성 중 14가지 (Duration, HeaderLength, IPversion header/payload, TTL, Proto, SrcIP, DstIP, SrcPort, DstPort, DIR, AckNum, WinSize, Timestamp)을 선정하고 추출하여 CSV 파일을 제작하는 프로그램을 구현하였다.

위의 Table 1을 중심으로 트래픽을 분석하였으며 생성된 CSV파일은 아래와 Fig. 3과 같다.

패킷 수집프로그램으로부터 수집된 CSV파일 데이터에 대한 분석을 진행하였다. 특징을 분석한 결과 기존의 특성을 이용하여 공격을 탐지할 경우 스캐닝 공격과 파일 전송을 탐지하기 어려웠으며 헤더 정보만을 가지고 분석하기에 한계가 있었다. 악성코드의 경우 한 번에 공격이 진행되는 것이 아닌 APT 또는 정상적인 접근을 이용하여 공격이 이뤄지고 window size의 경우도 다른 트래픽과의 차이가 없기 때문에 탐지가 불가능한 상황이 발생한다. 이를 해결하기 위해 본 연구에서는 트래픽의 특징을 분석하고 네트워크 특성을 선정하고 각 특성이 트래픽의 분류하는데 미치는 영향에 대한 성능 분석을 진행하였다. Table 1은 네트워크 트래픽을 분석하는데 사용되는 특성 중 많이 사용되는 특성을 정리한 표이다.

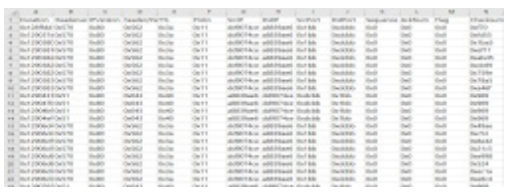

Fig. 3. CSV files created using data preprocessing

Table 1. 14 Features of Packet Data Characteristics

No.	Feature Name	Feature Description
1	Duration	Connection Time (seconds)
2	HeaderLength	header length of packet
3	IPversion	IPv6, IPv4 distinguished
4	header.Payload	IPv4, IPv6 header size
5	TTL	Packet Data Validity Period
6	Protocol	Protocol Information
7	SrcIP	Source IP
8	DstIP	Destination IP
9	SrcPort	Source port number
10	DstPort	Destination port number
11	Drop (dir)	Indicates whether the firewall has passed or deleted the packet, and indicates the direction of the packet
12	AckNum	Sequence response packet
13	WinSize	Data Division Size
14	Timestamp	A string that takes the elapsed time as a numeric value from a certain reference time (usually Epoch)

다음으로 수집한 데이터를 가지고 CNN을 이용하여 분류하기 위해 PNG 이미지로 특징을 정형화했다. 패킷 바이너리의 경우 크기가 모두 다르므로 크기가 커지면 그만큼 이미지의 세로 길이가 커진다. 트래픽을 시각화 할 경우 1바이트를 사용하기 때문에 256단계의 gray scale을 표현할 수 있다.생성된 이미지는 thumbnail로 변환해서 가로, 세로 길이에 상관없이 256x256 해상도로 변환했다.

파일의 HEX(0x00~0xFF)을 기준으로 높은 숫자는 진한 검정, 낮은 숫자는 옅은 회색으로 표현했다.

Fig. 4처럼 생성된 트래픽 특성을 이용한 이미지를 가지고 CNN 모델에 적용시켜 실험했다. 총 1100개(정상:1000, 비정상:100)의 바이너리를 가지고 정상: 비정상의 비율을 10:1로 실험했다. 훈련과 테스트에 사용한 데이터의 비율은 4대 1이다. learning_rate = 0.001, training_epochs = 15, batch_size = 2, 실험 모델의 Cost를 정의하

Fig. 4. Convert Thumbnail of Collected Data

기 위해 Softmax를 이용했으며, 최적화 도구로 "AdamOptimizer"로 사용했다.

Fig. 5는 CNN의 Architecture Structure이다. 이미지를 매번 디코딩하게 되면 머신러닝 모델의 학습단계에서 많은 입출력이 증가되게 된다. 시스템 처리 속도 향상을 위해 생성된 이미지를 128x128로 Resizing 작업을 진행하고 IDX파일을 생성하였다.

먼저 IDX 파일에서 128*128*3 크기의 트래픽 이미지를 읽는다. 이러한 이미지 픽셀은 [0, 255]에서 [0, 1]로 정규화 된다. 제 1 Convolutional layer(C1)은 크기가 3*3인 32개의 커널로 Convolutional 연산을 수행한다. C1 layer의 결과는 14*14 크기의 32개의 feature map을 이룬다. P1 layer에는 2*2 최대 Pooling 연산이 C1 layer에 이어진다. 제 2 Convolutional layer(C2)의 커널 크기도 3*3, 32 개의 채널을 구성한다. 결과는 7*7 크기의 32개의 feature map을 구성한다. 두 번째 2*2 최대 Pooling layer P2 이후 제 3 Convolutional layer(C3)의 커널 크기는 3*3, 64개의 채널을 갖는다. 결과는 4*4 크기의 64개의 feature map을 출력한다. Convolutional layer은 4 차원 텐서를 출력한다. 이제 Fully-Connected 네트워크에서 이를 입력으

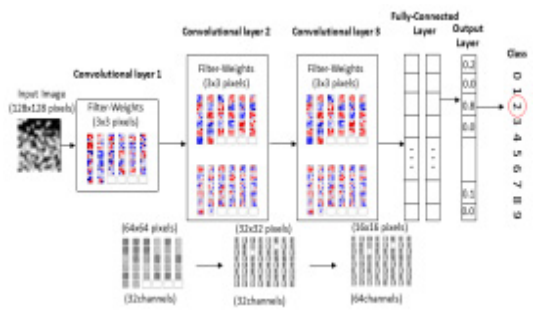

Fig. 5. CNN Architecture Structure

로 사용하고자 한다. 이 네트워크에서는 tensor가 2차원 tensor로 재구성되거나 평면화되어야 한다. 두 개의 layer는 Fully-Connected layer로써 결과 크기는 각각 128와 클래스 크기이다. softmax 함수는 각 클래스의 확률을 출력하는 데 사용된다. 게다가, 오버 피팅(over-fitting)을 줄이기 위해 드롭아웃(dropout)이 사용된다.

IV. 실 험

4.1 실험환경

TensorFlow는 Ubuntu 16.04 64비트 OS에서 실행되는 실험 소프트웨어 프레임워크로 사용된다. 서버는 8코어 및 16GB 메모리를 갖춘 i7-3770 CPU @ 3.40GHz이다. GeForce GTX 1060 6GB GPU가 가속기로 사용된다. 데이터는 총 40,000개의 데이터 중 Train은 31,920개, Test는 6080개이다. 저장된 디렉토리에 맞춰 class는 자동으로 나눠지게 된다.

4.2 실험 내용

시나리오 A(10 클래스 분류)는 트래픽 분류의 실제 적용에서 가장 기본적인 SRC_Port, DST_Port, DIR, Session Winsize, Timestamp을 이용한 악성코드에 의해 발생되는 트래픽을 식별하기 위한 특성에 따른 성능 실험을 진행하였다. 이 시나리오에서는 먼저 악성코드 또는 benign을 식별하기 위해 2진 분류가 수행된 다음 각 트래픽 클래스를 식별하기 위해 2개의 10 클래스 분류를 각각 수행했다. Fig. 6은 데이터 셋의 총 크기는 20,000개로 Train 셋이 16,800개, Test 셋이 3,200개이다. 각 트래픽은 10개의 class로 labeling 되어 있다. Train dataset에 대한 학습은 벡터의 성분을 tf.random API를 이용하여 랜덤하게 할당하였다.

Fig. 7은 시나리오 A의 Benign 분류 실험은 특성 6가지를 모두 사용하였을 때를 기준으로 실험을 진행하였으며 98.9% 정확도를 보였다.

Fig. 8은 Confusion Matrix 그래프[19]를 통해 "Gmail" 클래스와 "Weibo" 클래스의 오탐 비중이 상대적으로 높은 것을 확인할 수 있다. 유사 프로토콜 및 세션 처리과정이 비슷할 경우 오탐률이 증가하는 것을 확인할 수 있었다.

Fig. 9는 악성코드 분류 실험에 대한 결과이다. Validataion Accuracy는 98.0%의 성능으로 정상 트래픽을 분류하는 것을 볼 수 있었다. 상대적으로 Benign 분류 결과에 비해 0.9% 차이가 있었다.

Fig. 10은 Confusion Matrix 그래프를 통해 "Htbot" 클래스와 "Nsis-ay" 클래스의 오탐 비중이 상대적으로 높은 것을 확인할 수 있다. Nsis-ay의

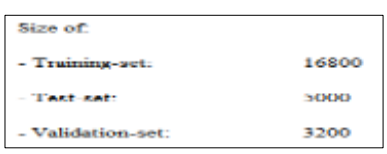

Fig. 6. Scenario A Dataset Size

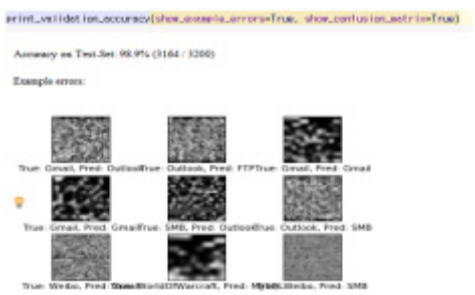

Fig. 7. Scenario A (Benign) Experimental Results

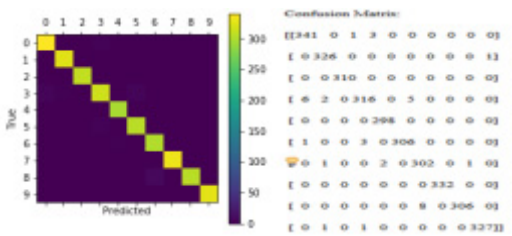

Fig. 8. Scenario A (Benign) Results Confusion Matrix

Fig. 9. Scenario A(Malware) Experimental Results

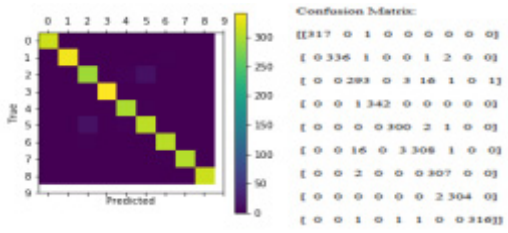

Fig. 10. Scenario A (Malware) Experiment Result Confusion Matrix

오탐율이 높은 이유는 sampling과정에서의 표본 데이터의 부족으로 인한 성능저하로 보인다.

Fig. 11은 시나리오 B(IoT 기기 트래픽 10종 분류)는 위의 실험결과를 바탕으로 SDN에서 수집한 IoT 기기 네트워크 트래픽을 실제 환경에서 발생한 트래픽을 분류한다. 특성은 시나리오 A와 같은 Src_Port, Dst_Port, DIR, Session Winsize, Timestamp을 이용하며, 특성을 감소시키며 실험을 진행하였다. 데이터 셋의 총 크기는 1,500개로 Train 셋이 1,260개, Test 셋이 240개이다.

Table 2.은 SDN에서 수집한 IoT 트래픽 클래스를 분류하였으며 아래와 같다.

Fig. 12는 시나리오 B의 각 공격 클래스 10개 Class에 대한 트래픽 분류 실험하였으며 다음과 같다.

시나리오 A에 비해 0.6~2.7%의 저하된 96.2%의 정확도를 볼 수 있었다. 성능저하의 원인은 학습 데이터 셋의 크기로 보았다. 실험에서 사용된 데이터

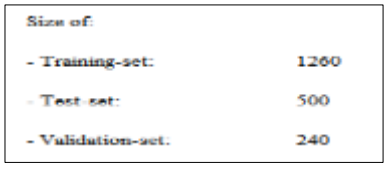

Fig. 11. Scenario B Dataset size

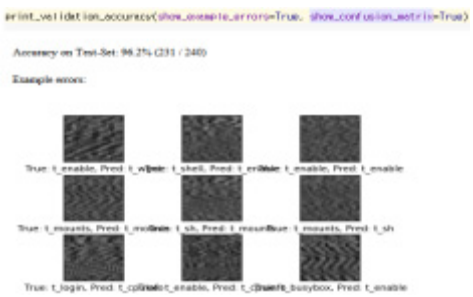

Fig. 12. Scenario B Experimental Results

Table 2. Traffic Class List

Count	Class name
1	login Suess, (Telnet)
2	login Fail (Telnet)
3	Enable commend
4	Shell Script Attack
5	IoT device Validation using Busy box Command
6	Mount Information
7	System Architecture cpuinfo Command
8	cat/bin/echo command using compile Architecture check
8	wget using download
10	CCTV benign Traffic

셋은 약 1,500개 불과하다. Fig. 13의 Confusion Matrix 그래프를 통해 "t_login" 클래스의 분류 정확도가 상대적으로 높은 것을 확인할 수 있다. "t_login" 클래스의 데이터 표본의 수가 전체 트래픽 데이터 수와 비슷하여 정확도가 높은 것으로 볼 수 있었다. 이는 데이터의 개수가 증가할수록 특징을 구성하는 개체 수가 많아져 침입 분석에 있어 보다 설득력 있는 정보의 추출이 가능함을 의미한다.

또한 V. Chawla는 딥러닝을 이용한 침입 탐지 연구에서는 클래스에 해당하는 데이터의 비율이 고르지 않은 경우 훈련 데이터 내 비율이 높은 클래스 쪽으로 결과를 내놓는 모델을 만들게 될 수 있다. 불균형의 정도가 심하거나 완만한 몇 가지 데이터셋에 SMOTEBoost를 적용한 결과에서 소수 클래스에 관한 예측 성능이 개선됨을 보였다[20].

Table 1의 다양한 데이터 셋의 특성을 대상으로 한 정확성을 분류하였다. 특성 중 악성코드를 분석 결과를 바탕으로 공통 특성 6가지를 선정하였으며 트래픽 분류 과정에서 각각 어떤 특성이 더 중요한지

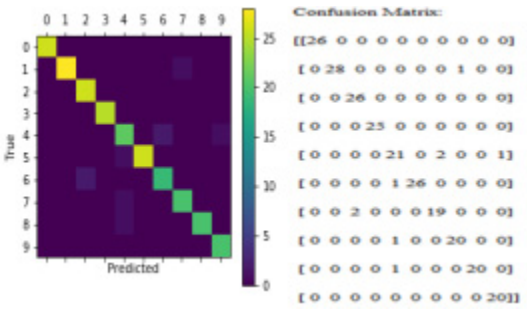

Fig. 13. Scenario B Results Confusion Matrix

Table 3. Classification Performance metrics vs Feature empolyed

Feature Set	Accuracy	F1	Precision	Recall
SRC_Port, DST_Port, DIR, Session Winsize,Timestamp	96.12	96.16	95.23	97.12
DIR, Session Winsize	96.52	95.85	95.38	96.33
Session Winsize	83.98	81.87	81.44	82.31

에 대한 실험을 진행하였다. Table 3은 각각의 특성에 따라 분류성능에 미치는 영향에 대해 분석하였다.

첫 번째 열은 훈련하는데 사용되는 특성을 제공한다. 오른쪽 열은 프로세스의 일반적인 성능을 나타낸다. 흥미롭게도 다양한 특성에 Timestamp의 특성의 경우 분류 성능에 대한 결과가 좋지 않았다. 또한 출발지 Port, 목적지 Port 특성이 분류에 미치는 영향이 적다는 것을 알 수 있었다. 하지만 DIR, Session, Winsize의 경우에는 분류 성능이 유지되었다. 이를 통해 기계학습을 이용한 트래픽 분류에서 특성에 의한 성능을 비교할 수 있었다. 논문에서 분석한 특성 중 port와 Timestamp의 경우 분류의 미치는 영향이 적으며 DIR, Session, Winsize의 경우 영향이 컸다.

V. 결 론

최근 ICT 기술의 발전과 IoT, 빅 데이터가 보급되며 많은 트래픽이 발생되고 있다. 또한 악의적인 공격에 의한 트래픽을 탐지하기 위한 다양한 연구가 진행되고 있지만 모든 트래픽을 탐지하기에는 역부족하다. 또한 관제에 소용되는 비용과 시간이 소요되며, 이런 손실을 줄이기 위해서 본 논문은 딥러닝을 이용한 비정상트래픽을 분류하기 위한 특성에 따른 성능평가를 진행하였다. 현재의 많은 연구들은 분류 성능 향상(정확도, 정밀성)을 향상시키기 위한 연구가 많이 진행되고 있다. 하지만 다양한 특성에 따라 달라지는 분류의 정확도의 추적하는 연구는 부족하다. 이를 해결하기 위해 본 연구에서는 트래픽의 특징을 분석하고 악성코드의 특성을 선정하여 특성 별 분류 성능에 미치는 영향에 대해서 분석하였다. 실험결과 세션만을 이용하였을 때보다 DIR, Session Winsize을 이용하였을 때 오탐율 줄이고 탐지율은 향상 시킬 수 있었다. 또한 IP와 Port의 경우 영향을 분류 성능에 영향을 많이 미치지 않는 것으로 분석되었다. 향후의 예측률을 향상시키기 위해 네트워크 트래픽의 데이터 불균형문제를 해결하기 위한 기계학습 기반 트래픽 생성에 대한 연구를 진행할 예정이다. 또한 유사 프로토콜 및 세션 처리과정을 가진 트래픽의 오탐율을 줄이기 위한 연구를 진행할 예정이다.

References

[1] A. Nordrum, "Popular Internet of Things Forecast of 48Billion Devices by 2020 Is Outdated", August 2016

[2] Alexander Khalimonenko, Oleg Kupreev, "DDOS attacks in Q1 2017", Securelist, 05. 2017

[3] Minn, Yin Pa, et al. "IoTPOT: Analysing the rise of IoT compromises." 9th USENIX Workshop on Offensive Technologies (WOOT). USENIX Association. 2015.

[4] MAL-FUQAHA, Ala, et al. Internet of things: A survey on enabling technologies, protocols, and applications. IEEE communications surveys & tutorials, 17(4): pp. 2347-2376. 2015,

[5] W. Haider, J. Hu, and M. Xie, "Towards reliable data feature retrieval and decision engine in host-based anomaly detection systems," in 2015 IEEE 10th Conference on Industrial Electronics and Applications (ICIEA), pp. 513-517. 2015,

[6] Xie, Miao, Jiankun Hu, and Jill Slay. "Evaluating host-based anomaly detection systems: Application of the one-class svm algorithm to adfa-ld." Fuzzy Systems and Knowledge Discovery (FSKD), 2014 11th

International Conference on. IEEE, pp. 978-982. 2014.
〔7〕 M. Guerroumi, A. Derhab, and K. Saleem, "Intrusion Detection System against Sink Hole Attack in Wireless Sensor Networks with Mobile Sink," in 2015 12th International Conference on Information Technology - New Generations, pp. 307-313. 2015,
〔8〕 Luo, Yuxuan, et al. "PU Learning in Payload-based Web Anomaly Detection." 2018 Third International Conference on Security of Smart Cities, Industrial Control System and Communications (SSIC). IEEE, pp. 1-5.2018
〔9〕 WANG, Wei, et al. Malware traffic classification using convolutional neural network for representation learning. In: Information Networking (ICOIN), 2017 International Conference on. IEEE, pp. 712-717.2017.
〔10〕 Li, Yuancheng, Rong Ma, and Runhai Jiao. "A hybrid malicious code detection method based on deep learning." International Journal of Security and Its Applications 9.5 pp.205-216.. 2015
〔11〕 KONG, Deyan, et al. A Big Network Traffic Data Fusion Approach Based on Fisher and Deep Auto-Encoder. Information. pp.2078-2489, 2016,
〔12〕 Tang, Tuan A., et al. "Deep learning approach for network intrusion detection in software defined networking." Wireless Networks and Mobile Communications (WINCOM), 2016 International Conference on. IEEE, pp. 258-26 2016.
〔13〕 W. Jung, S. Kim, and S. Choi, "Poster: Deep learning for zero-day ash malware detection," 2015.
〔14〕 Nataraj, L., Yegneswaran, V., Porras, P., & Zhang, J. (2011, October). A comparative assessment of malware classification using binary texture analysis and dynamic analysis. In Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence ACM .pp. 21-30. 2011
〔15〕 KIM, Jihyun, et al. An Approach to Build an Efficient Intrusion Detection Classifier. Journal of Platform Technology, 3.(4).pp. 43-52.2015,
〔16〕 KIM, Jihyun, et al. Long short term memory recurrent neural network classifier for intrusion detection. In: Platform Technology and Service (PlatCon), 2016 International Conference on. IEEE, pp. 1-5. 2016.
〔17〕 BEDIAKO, Peter Ken. Long Short-Term Memory Recurrent Neural Network for detecting DDoS flooding attacks within TensorFlow Implementation framework. 2017.
〔18〕 LIU, Yuchen; LIU, Shengli; ZHAO, Xing. Intrusion detection algorithm based on convolutional neural network. DEStech Transactions on Engineering and Technology Research, iceta. 2017,
〔19〕 BATISTA, Gustavo EAPA; PRATI, Ronaldo C.; MONARD, Maria Carolina. A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD explorations newsletter, 6.1: pp. 20-29. 2004,
〔20〕 V. Chawla, A. Lazarevic, L. O. Hall, and K. W. Bowyer, "SMOTEBoost: Improving prediction of the minority class in boosting," In European Conference on Principles of Data Mining and Knowledge Discovery, pp. 107-119, 2003.
〔21〕 Michael Collins.Network Security Through Data Analysis: From Data to Action.312.2014

〈저자소개〉

이 우 호 (Wooho Lee) 학생회원
2016년 2월: 순천대학교 정보통신학과 공학사
2018년 2월: 전남대학교 정보보안협동과정 이학석사
2018년 3월~현재: 전남대학교 정보보안협동과정 박사과정
〈관심분야〉 정보보호, 딥러닝, 네트워크 보안

노 봉 남 (Bongnam Noh) 종신회원
1978년: 전남대학교 수학교육과 학사
1982년: KAIST 대학원 전산학과 석사
1994년: 전북대학교 대학원 전산과 박사
1983년~현재: 전남대학교 전자컴퓨터공학부 교수
2000년~현재: 전남대학교 시스템보안연구센터 소장
〈관심분야〉 정보보안, 시스템 및 네트워크 보안

정 기 문 (Kimoon Jeong) 정회원
1999년 2월: 전남대학교 전산학과 학사
2001년 8월: 전남대학교 전산통계학과 석사
2009년 8월: 전남대학교 정보보안협동과정 이학박사
2001년 7월~2004년 12월: 한국정보보호진흥원
2004년 12월~2005년 6월: 국가사이버안전센터
2005년 6월~현재: 한국과학기술정보연구원 선임연구원
〈관심분야〉 딥러닝, 네트워크보안, 클라우드보안