J. Internet Comput. Serv.
ISSN 1598-0170 (Print) / ISSN 2287-1136 (Online)
http://www.jics.or.kr

오토 인코더 기반의 단일 클래스 이상 탐지 모델을 통한 네트워크 침입 탐지☆

Network Intrusion Detection with One Class Anomaly Detection Model based on Auto Encoder.

민 병 준[1] 유 지 훈[1] 김 상 수[2] 신 동 일[1] 신 동 규[1*]
Byeoungjun Min Jihoon Yoo Sangsoo Kim Dongil Shin Dongkyoo Shin

요 약

최근 네트워크 환경에 대한 공격이 급속도로 고도화 및 지능화 되고 있기에, 기존의 시그니처 기반 침입탐지 시스템은 한계점이 명확해지고 있다. 지능형 지속 위협(Adavanced Persistent Threat; APT)과 같은 새로운 공격에 대해서 시그니처 패턴은 일반화 성능이 떨어지는 문제가 존재한다. 이러한 문제를 해결하기 위해 기계학습 기반의 침입 탐지 시스템에 대한 연구가 활발히 진행되고 있다. 하지만 실제 네트워크 환경에서 공격 샘플은 정상 샘플에 비해서 매우 적게 수집되어 클래스 불균형(Class Imbalance) 문제를 겪게 된다. 이러한 데이터로 지도 학습 기반의 이상 탐지 모델을 학습시킬 경우 정상 샘플에 편향된 결과를 가지게 된다. 본 논문에서는 이러한 불균형 문제를 해결하기 위해서 오토 인코더(Auto Encoder; AE)를 활용해 One-Class Anomaly Detection 을 수행하여 이를 극복한다. 실험은 NSL-KDD 데이터 셋을 통해 진행되었으며, 제안한 방법의 성능 평가를 위해 지도 학습된 모델들과 성능을 비교한다.

☞ 주제어 : 이상 탐지, 네트워크 침입 탐지, 오토인코더, NSL-KDD

ABSTRACT

Recently network based attack technologies are rapidly advanced and intelligent, the limitations of existing signature-based intrusion detection systems are becoming clear. The reason is that signature-based detection methods lack generalization capabilities for new attacks such as APT attacks. To solve these problems, research on machine learning-based intrusion detection systems is being actively conducted. However, in the actual network environment, attack samples are collected very little compared to normal samples, resulting in class imbalance problems. When a supervised learning-based anomaly detection model is trained with such data, the result is biased to the normal sample. In this paper, we propose to overcome this imbalance problem through One-Class Anomaly Detection using an auto encoder. The experiment was conducted through the NSL-KDD data set and compares the performance with the supervised learning models for the performance evaluation of the proposed method.

☞ keyword : Anomaly Detection, Network Intrusion Detection, AutoEncoder, NSL-KDD

1. 서 론

최근 정보 통신 기술들의 발전에 따라 네트워크 환경의 규모는 매우 빠른 속도로 확장되었으며, 동시에 네트워크 환경에 대한 사이버 위협 또한 증가하기 시작하였다. 이러한 네트워크상에서의 사이버 위협을 탐지하기 위하여 대다수의 기업들은 네트워크 침입탐지 시스템(Network based Intrusion Detection System, NIDS)을 운용하고 있으며, 다양한 보안 공격이 발생하였을 경우 이를 관리자에게 보고하는 것을 목표로 한다. 기존에 운용되던 침입 탐지 시스템들은 오용 탐지(Misuse Detection) 방식으로, 시그니처 기반의 탐지 방법(Signature based Detection)을 주로 사용해 왔다. 이는 보안 전문가를 통해 이미 빈번히 사용되는 공격들에 대해 패턴을 정의해둔 것으로, 입력된 트래픽과 비교를 통해서 공격을 탐지한다. 하지만 최근 APT(Advance Persistent Threat) 공격과 같이 위협이 고도화됨에 따라서 기존의 시그니처 기반의 탐지 방법은 알려진 공격 이외에는 탐지할 수 없다는 한계에 직면한다[1]. 또한 시그니처를 생성해내기 위한 시간과 비용적

1 Dept. of Computer Science, Sejong University, Region(city), 05006, Korea
2 Agency for Defense Development, Daejeon 305600, South Korea
* Corresponding author (shindk@sejong.ac.kr)
[Received 16 November 2020, Reviewed 30 November 2020, Accepted 17 December 2020]
☆ 본 연구는 국방과학연구소의 지원으로 수행되었습니다(위탁연구계약번호:UD200014ED)

문제 또한 발생한다.

비정상 행위 탐지(Anormaly Detection)방법은 기존의 오용 탐지 방법과 달리 정상 행위(Normal Behavior)에 대해 모델링을 통해 비정상 행위(Anomaly Behavior)를 탐지하는 방법으로, 알려지지 않은 공격(Zero-day Attack)에 대해서도 탐지 할 수 있다. 하지만 정상 행위에 대한 모델링은 간단한 문제가 아님에 따라서 최근 기계 학습(Machine Learning)기법을 도입해 이러한 문제를 해결하려는 연구가 활발히 진행되고 있다[2]. 기계 학습은 데이터로부터 모델링이 가능하며, 이를 통해 예측 결과를 추론할 수 있기에 이상 탐지에 적합하다. 기계학습에서 이러한 정상과 비정상을 구분 짓기 위한 문제는 이진 분류(Binary Classification)문제로 정의 될 수 있다.

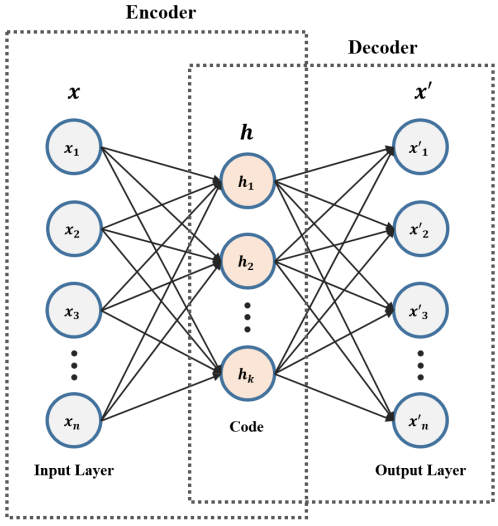
하지만 현실세계에서 발생하는 많은 데이터들은 각 클래스별로 불균형하게 데이터들이 분포하게 되는데, 이는 소수 클래스(Minor Class)의 데이터가 다수 클래스(Major Class)에 비해서 현저히 적은 데이터로 구성되는 것을 의미하며, 이를 불균형 데이터(Imbalanced Data)라고 부른다. 특히 침입 탐지 문제에서는 이러한 침입 데이터의 비중이 전체 데이터 중 약 1%로 알려져 있어[3], 이러한 불균형 데이터로부터 기계 학습 모델을 학습시켜야 되는 문제에 빠지게 된다. 따라서 일반적으로 많이 사용하는 지도 학습 (Supervised Learning) 기반의 모델을 불균형 데이터 셋에 적용할 경우 분류 성능의 저하를 야기할 수 있으며[4], 특히 소수 클래스들의 탐지율이 크게 저하되는데 이는 결정 경계(Decision Boundary)가 다수 클래스에 편향되도록 학습이 되기 때문이다[5].

본 논문에서는 실제 네트워크 환경에서 빈번히 발생하는 불균형 데이터에 무관하며, 새로운 공격들에 대해서도 탐지하기 위해서, 비지도 학습(Unsupervised-Learning) 모델인 오토 인코더(Auto-Encoder)를 기반으로 하는 One Class Anomaly Detection 모델을 제안한다. One-Class 학습 방법은 특정 클래스의 샘플만을 학습하는 방법으로 준지도 학습(Semi-supervised Learning)에 해당한다[6]. 이는 대부분의 샘플이 정상 샘플에 해당하는 침입 탐지 환경에서 매우 적합하다고 할 수 있다. 따라서 오토 인코더 모델을 정상 샘플만을 통해 학습시킨 뒤, 재구성 손실(Reconstruction Error)을 통해 공격 행위를 탐지할 것을 제안한다. 실험에 사용된 침입 탐지 데이터 셋은 NSL-KDD 데이터 셋을 사용하였으며, 결과는 지도 학습기반 모델들과 비교한다.

2. 관련 연구

2.1 오토 인코더

오토 인코더는 입력값과 출력값을 동일한 값으로 근사하는 비지도 학습 신경망(Unsupervised Neural Network) 모델이다[8]. Figure. 1과 같이 인코더(Encoder)와 디코더(Decoder)로 나뉘어 구성되며, Code 계층을 기준으로 대칭의 구조를 가진다. 중간에 위치하는 병목 구간(Bottelneck)이라 불리는 Code 계층은 인코더의 출력으로써, 입력 데이터의 저 차원의 잠재 공간으로 매핑된 결과를 출력한다. 디코더는 해당 벡터를 다시 입력으로 사용하여 입력 데이터를 재구성(Reconstruction)하는 과정을 통해 학습을 진행한다. 이러한 학습 과정에서 인코더는 복원을 위해 중요한 정보들을 최대한 보존하는 것을 목표로 압축을 시도하게 되며, 결국 중요한 핵심 정보들만을 인코딩 할 수 있게 된다.



(그림 1) 오토인코더의 구조
(Figure 1) Architecture of AutoEncoder

인코더는 식 1과 같이 정의되며, x는 입력값이고 W_e와 b_e는 인코더의 파라미터로 선형 결합 되어 활성 함수(activation function) σ의 입력으로 들어가게 된다. 활성함수 σ는 비선형 함수와 선형 함수를 사용할 수 있으며, 선형 함수를 사용할 경우 선형 특징 추출 기법인 PCA (Principal component analysis)와 유사하게 작동하게 된다.

h는 인코더의 출력에 해당하며, Code라 부르기도 한다. 디코더는 식 2와 같이 정의되며, 입력 값으로 h를 사용하여 입력 값을 재구성한 x'를 출력한다.

$$h = \sigma(W_e x + b_e) \tag{1}$$

$$x' = \sigma(W_d h + b_d) \tag{2}$$

오토 인코더의 손실 함수 L는 식 3과 같으며, 디코더의 출력 x'와 입력값 x의 MSE(Mean Squared Error)을 의미하며, 재구성 오류(Reconstruction Error)라고 불린다. 이 외에도 MAE(Mean Absolute Error)와 Cross Entropy Error 등이 사용될 수 있으며, 손실 함수 L을 최소화 하는 것을 목표로 학습을 진행한다. 적층 오토인코더(Stacked Auto Encoder)는 이러한 오토 인코더가 여러 은닉층을 가지는 모델로 더욱 깊은 구조를 가질수록 복잡한 데이터에 대해서 학습할 수 있다.

$$L(x', x) = \frac{1}{N}\sum_{i=1}^{N}(x' - x)^2 \tag{3}$$

2.2 기계학습 기반의 침입 탐지 시스템

기계 학습기반의 침입 탐지 연구가 활발히 진행되고 있으나, 불균형 데이터로부터 학습하는데 여전히 어려움을 겪고 있다. Yanqing Yang et. al. [10]은 이러한 문제를 해결하기 위해서 생성모델 ICVAE (Improved Conditional Variational AutoEncoder)를 제안하였으며, 이를 통해 데이터 불균형을 데이터를 해소하여 심층 신경망 모델을 학습시켰다. Javaid et al. [11]은 딥 러닝 접근법인 STL (Self-taught Learning) 방법을 통해 네트워크 침입 탐지 시스템을 제안하였다. 해당 모델은 2가지 스텝으로 나뉘어 진행되며, Sparse Auto Encoder를 학습시킨 뒤, 학습된 인코더의 출력을 통해 분류기를 학습시키는 방식이다. Kim et al. [12]는 SVM(Support Vector Machine) 기반의 침입 탐지 시스템을 개선하기 위해서 GA(Genetic Algorithm)과 융합하여 사용하는 것을 제안하였다. 이를 통해 최적 파라미터를 선택하는 것뿐 아니라 최적 특징 셋 또한 선택할 수 있게 되었다고 보고하고 있다. Yin et al. [13]은 RNN(Recurrent Neural Network) 모델을 활용한 침입 탐지 시스템을 제안하였다. 은닉층 파라미터 수에 따른 학습 결과를 상세히 보고하고 있으며, 이진 분류뿐 아니라 다중 분류에 대한 실험 결과도 보고하고 있다.

3. 연구 방법

3.1 제안하는 방법

본 논문에서는 One-Class Anomaly Detection based Auto Encoder 방법을 제안한다. One-Class 학습 방법은 특정 종류의 클래스 데이터만을 가지고 학습시키는 방법으로, 네트워크 침입 탐지 문제와 같이 클래스 불균형이 일반적인 상황에서 적용한다. 따라서 학습에 사용된 오토 인코더는 Normal-Class 데이터들만을 사용하여 재구성 오류를 최소화 하도록 훈련된다. Normal-Class 샘플의 특징들에 대해서만 오토 인코더가 학습을 진행하였음에 따라서 학습에 사용되지 않은 샘플들에 대해서는 비교적 높은 오류 값들의 분포를 가질 것으로 기대 할 수 있다.

일반적으로 오토 인코더는 학습이 진행 된 이후 인코더 네트워크를 통해 특징 추출(Feature Extraction)을 목적으로 사용하지만, 본 논문에서는 재구성 오류 값을 통한 이상 탐지를 제안하였음에 따라서 디코더 네트워크를 제거하지 않고 전체 학습된 오토인코더 네트워크를 그대로 사용한다. 학습된 오토 인코더는 새로운 입력 데이터가 학습된 데이터(Normal Data)와 유사할 경우 비교적 낮은 손실 값을 가지게 되며, 그와 동떨어진 데이터가 입력으로 사용될 경우 재구성 손실 값이 크게 나타나는 것을 전제로 한다. 따라서 정상 데이터만 학습한 오토인코더는 비정상 행위에 대해서 높은 손실 분포를 가지게 된다. 본 논문에서는 이러한 재구성 손실 값을 통해 적절한 임계값을 찾고 정상과 공격을 구분 짓는 방법을 제안한다. 임계값은 θ로 명시하며, 임계값 θ보다 큰 재구성 손실의 결과는 공격으로 예측하게 된다. 이후 모델의 적절한 임계값 θ를 결정하는 것을 목표로 하는데, [7]에서는 학습에 사용된 Train 데이터 셋들의 재구성 손실로부터 백분위 값을 통해 임계값 θ를 결정하는 방법을 소개하고 있다. Test 데이터 셋으로부터 직접 임계값 θ를 계산하는 것은 바람직하지 않음에 따라서 본 논문에서 또한 이러한 학습 데이터의 재구성 손실 값에 따른 백분위수를 이용하여 θ값을 결정한 뒤, 테스트 결과에서 가장 좋은 모델을 사용한다.

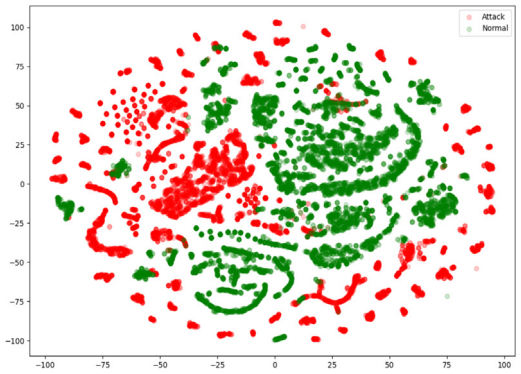
3.2 NSL-KDD 데이터 셋

NSL-KDD 데이터 셋은 1999년 DARPA 침입탐지 평가 프로그램을 통해 만들어진 KDD CUP 99 데이터 셋을 M.Tavallaee et. al. [9] 가 개선하여 제안한 데이터 셋으로,

미 공군의 네트워크를 모델링하여 38가지의 네트워크 침입 탐지 공격 시뮬레이션을 통해 만들어 졌다. M. Tavallaee는 KDD CUP 99 데이터 셋의 규모가 지나치게 크며, 많은 중복 레코드 등을 포함하는 문제점이 있다고 지적하였다. 이는 발생 빈도가 높은 공격에 데이터가 매우 치우쳐져 있음을 의미한다. 사용된 38 공격 기법은 Table 1에서 제시하는 4개의 공격 유형으로 분류할 수 있다.

(표 1) NSL-KDD 데이터 셋의 공격 유형
(Table 1) Attack type of NSL-KDD Dataset

Type	Description
Normal	normal traffic
DoS	Denial of Service
Probe	Pre-operation for vulnerability analysis before intrusion
U2R	Unauthorized access to take over root authority
R2L	Attempting unauthorized access from remote



(그림 2) t-SNE 기법을 통한 NSL-KDD 시각화
(Figure 2) Visualization using t-SNE for NSL-KDD

본 논문에서는 전체 공격에 대해서 이상 탐지를 목표로 하기에 Table 1에서 제시되는 4가지의 공격 유형을 모두 1가지 공격으로 간주한다. 따라서 클래스 이름을 모두 Attack으로 통일하며, Normal과 Attack을 분류하는 것을 목표로 한다. Figure 2는 NSL-KDD Train 셋에서 무작위로 추출된 2만개의 샘플들을 t-SNE (t-Distributed Stochastic Neighbor Embedding) 기법을 통해 시각화 한 그림이다. 이를 통해 정상 샘플들과 공격 샘플들이 선형 분리가 불가능 한 것을 확인할 수 있으며, 정상 샘플들로부터 비선형 관계들을 모델링 가능한 학습모델을 사용해야 함을 알 수 있다. 본 논문에서 실험에 사용되는 학습 모델들은 모두 비선형 분리가 가능한 모델들로 구성되며, 오토 인코더의 활성함수 또한 선형 함수가 아닌 비선형 함수로 사용해야 정상 데이터들에 대한 올바른 모델링이 가능한 것으로 확인된다.

3.3 데이터 셋 전처리

3.3.1 불필요한 특징 제거

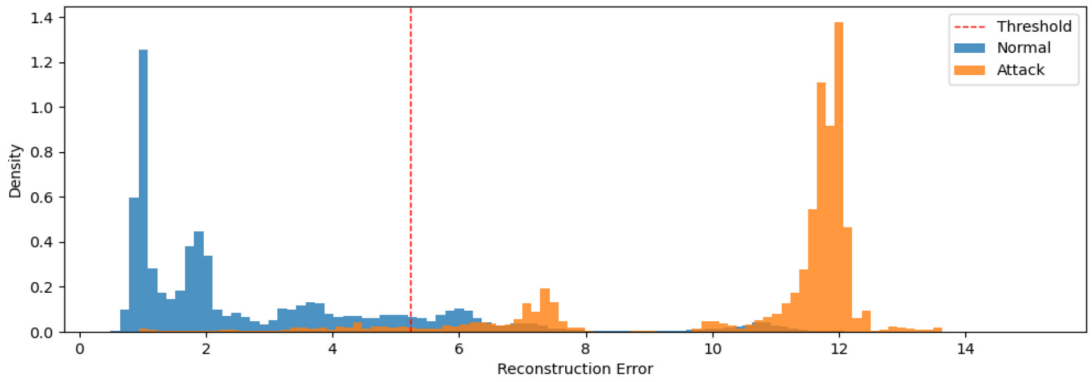
NSL-KDD 데이터 셋은 42개의 input feature들로 구성되어 제공되며, 이중 difficulty 속성은 학습과 무관하여 사전에 제거하였으며, num_outbound_cmds 속성은 데이터를 살펴본 결과 표준 편차가 0으로 모든 데이터의 값이 동일한 것으로 확인되어 사전에 제거하였다. 이로써 40개의 input feature dimenstion 으로 최초 구성되어 전처리를 진행한다.

3.3.2 데이터 정규화

3.3.1절을 통해 불필요한 feature들을 사전에 제거한 뒤, 데이터 정규화(Normalization)과정을 통해서는 최종적으로 모든 feature 들을 0과 1사이의 값으로 변경하는 것을 목표로 한다. 전처리는 데이터 형식에 따라 달리 진행하였으며, NSL-KDD 데이터 셋의 데이터 형식은 nominal, numeric, binary 3가지로 구분 지을 수 있다. nominal type 데이터들은 범주형 문자 데이터들로 신경망의 입력으로 사용할 수 없는 형태이다. 따라서 모두 정수형으로 인코딩 한 뒤 one-hot 벡터로 변환하였다. nominal type 데이터들의 one-hot vector 표현에 따라 데이터의 입력 차원이 크게 증가하였다.

$$x_{new} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{4}$$

numeric type 데이터들에 대해서는 속성 값들의 범위의 차이를 왜곡하지 않고 공통 스케일로 변경하기 위해 식 4와 같이 최소 최대 정규화(Min-max Normalization)를 진행하였으며, binary type데이터들의 경우 모두 0과 1로 구성되기 때문에 별다른 전처리 과정을 수행하지 않았다. 이를 통해 40차원의 feature dimension에서 121차원의 feature dimension으로 최종 변환되어 모델의 입력으로 사용하게 된다.



(그림 3) 학습된 적층 오토 인코더의 재구성 손실 분포
(Figure 3) Reconstruction Loss Distribution of trained Stacked Auto Encoder

4. 실 험

본 실험에서는 오토인코더의 모델 구조 변화에 따른 실험을 진행하였으며, 추후 지도학습 기반의 모델들과 성능을 비교하였다. 각 모델들의 분류를 위한 임계값 θ는 3.1.1 절에서 언급한 학습 데이터로 사용된 Normal-Class 샘플들의 재구성 손실의 백분위 값을 이용하여 설정하였다. 학습에 사용된 데이터는 Table 2와 같이 KDDTrain+ 데이터 셋과 KDDTest+ 데이터 셋을 사용하였다. One-Class Learning을 진행하였기에 KDDTrain 데이터 셋에는 공격 데이터의 샘플들을 모두 제거하여 모델의 학습에 사용하였다. 실험에 사용된 오토인코더 모델은 Code 계층을 기점으로 좌우 대칭한 형태로 구성되며, 인코더와 디코더의 각 은닉층은 Code 크기로부터 512까지 배수로 늘어나며 구성된다. 이외 학습에 사용된 파라미터 값은 Table 3과 같다.

(표 2) 실험에 사용된 NSL-KDD 데이터 셋
(Table 2) NSL-KDD Dataset for Experiment

Class	KDD Train+	KDD Test+
Normal	67343	9711
Attack	0	12833

(표 3) 실험에 사용된 학습 파라미터
(Table 3) Experiment Parameters for Train

Auto-Encoder Parameters	
Model Layer	121 (input, output) 512 - 256 - ... - code (encoder) code - ... - 256 - 512 (decoder)
Activation Function	Leaky Relu(hidden), Relu(output)
Dropout	0.5
Batch size / Epoch	64 / 10
Optimizer / Learning rate	Adam / 0.001
Loss Function	Mean Squared Error

4.1 성능 평가 지표

본 연구에서는 모델의 성능을 평가하기 위해 혼동 행렬(Confusion Matrix)을 사용한다. 오차행렬을 통해서 정확도(Accuracy), 정밀도 (Precision), 재현율(Recall), F1 스코어(F1 score)를 계산하고 이를 바탕으로 모델별 성능을 비교 분석한다. 혼동 행렬은 데이터 셋에 대해 모델이 분류한 결과를 나타내는 표로, 네트워크 이상 징후 탐지 모델의 오차행렬은 Table 4와 같이 나타낼 수 있다.

(표 4) 혼동 행렬
(Table 4) Confusion Matrix

		Actual	
		Positive	Negative
Predicted	Positive	TP	FP
	Negative	FN	TN

정확도는 전체 샘플 중 맞게 예측한 샘플 수의 비율을 뜻하며, 높을수록 좋은 모형이다. 전체 데이터 셋에서 정상과 공격 트래픽을 올바르게 예측한 비율에 해당한다. 정확도는 분류 모델의 성능을 평가하는데 일반적으로 사용되는 지표이지만, 불균형 데이터 셋에 대해서는 왜곡된 결과를 야기한다. 따라서 이러한 문제점을 보조하기 위한 지표로 정밀도(Precision)와 재현율(Recall) 그리고 F1 Score를 보조 지표로 사용한다.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

$$\mathrm{Precision} = \frac{TP}{TP+FP} \quad (6)$$

$$Recall = \frac{TP}{TP+FN} \quad (7)$$

$$F1\ \ Score = 2* \frac{\mathrm{Precision} \times Recall}{\mathrm{Precision} + Recall} \quad (8)$$

정밀도(Precision)은 양성 클래스에 속한다고 출력한 샘플 중 실제로 양성 클래스에 속하는 샘플 수의 비율을 말하며 식 6과 같다. 재현율(Recall)은 실제 양성 클래스에 속한 표본 중에 양성 클래스에 속한다고 출력한 표본의 수의 비율을 뜻하며 식 7과 같다. 정밀도와 재현율은 상호 보완적인 평가 지표이므로, 어느 한쪽의 수치를 강제로 높이면 다른 한쪽의 수치는 떨어질 가능성이 있다. F1 Score는 정밀도와 재현율의 조화 평균을 의미하며, 불균형 클래스에서 정확한 평가를 위해 주로 사용되는 지표로 식 8과 같다.

4.2 오토 인코더 기반 네트워크 이상 탐지 실험

오토 인코더의 Code계층의 크기는 하이퍼 파라미터(Hyper-parameter)로 오토 인코더의 복원 성능에 영향을 준다. 따라서 본 실험에서는 Code의 크기 값을 8부터 64까지 배수로 변화시키며 모델을 각각 구성하였다. 이에 따라 Table 5와 같이 4가지 구조의 적층 오토 인코더 모델이 만들어지며, 각 모델들의 성능을 비교하였다. 실험은 KDD Test+ 데이터 셋을 통해 진행되었으며, 결과는 Table 5와 같다. 4가지 병목 구간의 크기를 가진 모델 모두 90%에 근접하는 지표들을 출력하는 것을 확인할 수 있었으며, 본 실험에서는 16 크기의 병목 구간을 가지는 적층 오토 인코더 모델이 가장 좋은 성능을 보이는 것을 알 수 있었다.

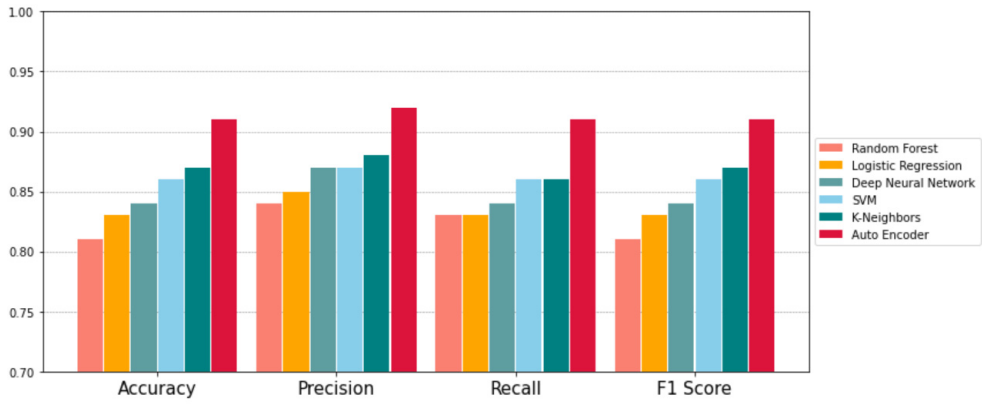
Figure 3은 Table 5에서 가장 좋은 성능을 보인 16 병목 구간을 가지는 적층 오토 인코더 모델을 통해 KDD Train+ 데이터 셋에 대한 재구성 손실 분포를 히스토그램으로 시각화한 것으로, x축은 재구성 손실 값으로 구성되며, y축은 손실 값의 밀도에 해당한다. 좌측에 위치한 분포는 Normal-Class의 분포이며, 우측에 분포는 Attack-Class의 재구성 손실 분포로 두 분포가 매우 동떨어진 것을 확인할 수 있다. 이를 통해 본 논문에서 제안한 One-Class Anomaly Detection based Auto Encoder 방법이 네트워크 트래픽을 손실 값을 통해 분류하는데 있어 매우 효과적임을 알 수 있으며, 손실 값들로부터 임계값을 사용하면 두 클래스를 Figure 3과 같이 분류할 수 있음을 알 수 있다. Table 5의 분류 결과는 Figure 3에 표시된 임계값을 사용한 결과로 θ값은 5.2278값에 해당하며, 학습 손실 값들 중 82 백분위수에 해당하는 값이다.

(표 5) Code 파라미터에 따른 오토인코더의 성능 비교
(Table 5) Performance Comparison of Auto-Encoder by Code value.

Code	Accuracy	Precision	Recall	F1 Score
8	0.90	0.90	0.90	0.90
16	0.91	0.92	0.91	0.91
32	0.89	0.89	0.89	0.89
64	0.91	0.91	0.91	0.91

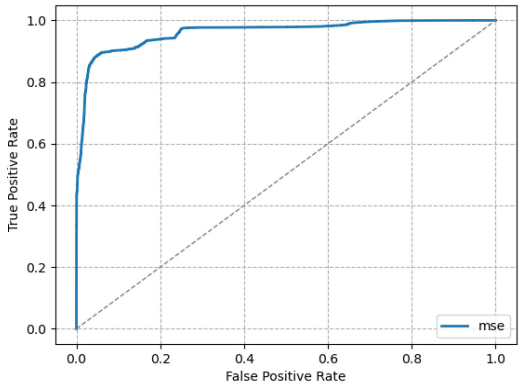
(표 6) 지도학습 모델들과의 실험 결과 비교
(Table 6) Experiment Results compared with Supervised Model

Model	Accuracy	Precision	Recall	F1 Score
Random Forest	0.81	0.84	0.83	0.81
Logistic Regression	0.83	0.85	0.83	0.83
Deep Neural Network	0.84	0.87	0.84	0.84
SVM	0.86	0.87	0.86	0.86
K-Neighbors	0.87	0.88	0.87	0.87
Stacked Auto Encoder (16)	0.91	0.92	0.91	0.91



(그림 4) 오토 인코더와 지도 학습 모델들의 성능 비교
(Figure 4) Performance comparison between Auto Encoder and Supervised Learning models.

Figure 5는 16 병목 구간 모델의 수신자 조작 특성 곡선(Receiver Operator Characteristic Curve; ROC Curve)을 나타낸다. 수신자 조작 특성 곡선은 이진 분류 시스템에 대한 성능 평가 기법으로 자주 사용되며, 분류에 좋은 모델의 경우 외측 상단에 가까운 곡선의 형태를 그리게 된다. 이를 정량적으로 평가하기 위한 방법으로 AUC(Area Under Curve)값을 사용할 수 있는데 이는 수신자 조작 특성 곡선의 아래 면적을 의미하며, 이를 AUROC라고 부른고 최대값을 1로 가진다. 본 실험에서 가장 좋은 성능을 낸 모델의 AUROC 값은 0.962의 값을 가지며, 이는 모델이 매우 좋은 분류 성능을 보이는 것을 알 수 있다.



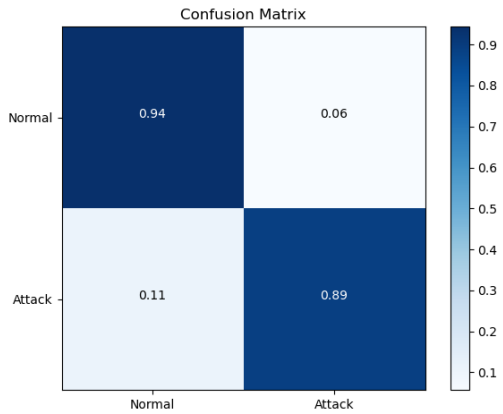
(그림 5) 오토인코더의 수신자 조작 특성 곡선
(Figure 5) ROC Curve for Auto Encoder

Table. 6과 Figure 4는 Table 5의 표시된 실험에서 가장 좋은 성능을 보인 적층 오토 인코더 모델과 지도 학습 모델들의 성능을 비교한 결과이다. 해당 모델들은 지도 학습 방법으로 학습되었음에 따라, Normal 데이터뿐만 아니라 Attack 데이터들 또한 포함하여 모든 데이터 셋을 학습하였다. 트리 기반의 앙상블 모델인 Random Forest 모델은 최근까지도 많이 사용되고 있는 모델이지만, 현재 주어진 실험에서는 좋은 결과를 보이지 못하였다. 또한 현재 많은 도메인에서 사용되고 있는 심층 신경망 분류기 모델은 Random Forest 모델과 비교해 제법 우수한 성능을 보이고 있지만, 그럼에도 불구하고 본 논문에서 제시한 모델의 성능에는 미치지 못하는 것을 알 수 있다. 기존의 지도학습 모델들 중에서는 KNN(K-Neighbors) 분류기 모델이 가장 좋은 성능을 보이고 있음을 알 수 있었다. 이들 모델들을 F1 Score를 기준으로 비교시 본 논문에서 제시한 모델은 가장 성능이 낮은 모델과 비교해 약 10%의 성능 차이가 나는 것을 확인할 수 있다.

(표 7) 제안하는 모델의 세부 분류 결과
(Table 7) Detailed Classification Results for Proposed Models

Code	Precision	Recall	F1 Score	Support
Normal	0.87	0.94	0.90	9711
Attack	0.95	0.89	0.92	12833
Total	0.92	0.91	0.91	22544

Table 7은 본 논문에서 제안한 모델의 클래스별 세부 분류 결과이다. 정밀도(precision)과 재현율(recall)값은 분류 결정 임계값에 영향을 받으며, 둘의 관계는 trade off 관계이다. 한쪽 지표가 과도하게 높을 경우 문제가 되지만, 본 논문에서의 모델은 두 지표 값이 모두 높은 것을 확인할 수 있다. 최종적으로 학습된 모델의 결과를 혼동 행렬로 분석한 결과는 Figure. 6과 같으며, 정규화된 결과치이다. 분류 대상 클래스들에 대해서 높은 수치를 보이고 있음을 알 수 있다.



(그림 6) 제안된 모델의 이상 탐지 혼동 행렬
(Figure 6) Anomaly detection confusion matrix for proposed models

5. 결 론

본 논문에서는 오토 인코더(Auto Encoder)모델을 통한 One-Class Anomaly Detection 모델을 제안하여 네트워크 침입 탐지에 대한 연구를 진행하였다. 실제 네트워크 침입 데이터는 매우 불균형함에 따라 기존의 지도 학습 기반의 분류 방법은 적합하지 않음을 알 수 있었으며, 다수 클래스에 해당하는 Normal-Class 데이터만을 통해 학습한 오토 인코더 기반의 이상 탐지 모델이 매우 좋은 성능을 보이는 것을 실험을 통해 확인하였다. 또한 적층 오토 인코더의 재구성 손실로부터 공격 행위를 탐지하는 방법에 대해서 서술하였으며, 학습에 사용된 샘들의 재구성 손실만을 사용하여 적합한 임계값을 찾을 수 있음을 보였다. 적층 오토 인코더의 최적의 모델 구조를 찾기 위해서 병목 구간의 크기를 변경하며, 모델의 성능을 평가하였으며 16 사이즈를 가지는 모델이 가장 좋은 성능을 가지는 것을 확인할 수 있었다. 이후 비교군 모델들과의 성능 비교에서는 F1 score를 기준으로 가장 낮은 탐지 모델은 81%의 성능을 보여주었으며, 본 논문에서 제시한 모델은 10%나 성능이 개선된 91% 우수한 성능을 보임을 알 수 있었다.

참고문헌(Reference)

[1] M. Thottan and C. Ji, “Anomaly detection in IP networks”, IEEE Transactions on signal processing, vol. 51, no. 8, pp. 2191-2204, 2003. https://doi.org/10.1109/tsp.2003.814797

[2] M. Ahmed, A. N. Mahmood and J. Hu, “A survey of network anomaly detection techniques”, Journal of Network and Computer Applications, vol 60, pp. 19-31, 2016. https://doi.org/10.1016/j.jnca.2015.11.016

[3] J. Song, H. Takakura, Y. Okabe and Y. Kwon, “Correlation analysis between honeypot data and IDS alerts using one-class SVM”, Intrusion Detection Systems, pp. 173-192, 2011. https://doi.org/10.5772/13951

[4] R. Longadge and S. Dongre, “Class imbalance problem in data mining review”, 2013. Preprint at https://arxiv.org/abs/1305.1707

[5] S. Barua, M. M. Islam, X. Yao and K. Murase, “MWMOTE--majority weighted minority oversampling technique for imbalanced data set learning”, IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 2, pp. 405-425, 2012. https://doi.org/10.1109/tkde.2012.232

[6] L. M. Manevitz and M. Yousef, “One-class SVMs for document classification”, Journal of machine Learning research, vol 2, pp. 139-154, 2001. https://dl.acm.org/doi/10.5555/944790.944808

[7] A. Borghesi, A. Bartolini, M. Lombardi, M. Milano and L. Benini, “Anomaly detection using autoencoders in high performance computing systems”, In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 9428-9433, 2019. https://doi.org/10.1609/aaai.v33i01.33019428

[8] T. Luo and S. G. Nagarajan, “Distributed anomaly detection using autoencoder neural networks in wsn for

iot", IEEE International Conference on Communications (ICC), pp. 1-6, 2018.
https://doi.org/10.1109/icc.2018.8422402

[9] M. Tavallaee, E. Bagheri, W. Lu and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set", IEEE symposium on computational intelligence for security and defense applications, pp. 1-6, 2009.
https://doi.org/10.1109/cisda.2009.5356528

[10] Y. Yang, K. Zheng, C. Wu and Y. Yang, "Improving the classification effectiveness of intrusion detection by using improved conditional variational autoencoder and deep neural network", Sensors, vol. 19, no. 11, pp. 2528, 2019.
https://doi.org/10.3390/s19112528

[11] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system", In Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS), pp. 21-26, 2016.
https://dl.acm.org/doi/10.4108/eai.3-12-2015.2262516

[12] D. S. Kim, H. N. Nguyen and J. S. Park, "Genetic algorithm to improve SVM based network intrusion detection system", In 19th International Conference on Advanced Information Networking and Applications (AINA papers), vol. 2, pp. 155-158, 2005.
https://doi.org/10.1109/aina.2005.191

[13] C. Yin, Y. Zhu, J. Fei and X. He, "A deep learning approach for intrusion detection using recurrent neural networks", Ieee Access, vol. 5, pp. 21954-21961, 2017.
https://doi.org/10.1109/access.2017.2762418

◐ 저 자 소 개 ◑

민 병 준(Byeongjun Min)
2019년 세종대학교 대학원 컴퓨터공학과 (공학석사)
2019년~현재 세종대학교 대학원 박사과정
관심분야 : 강화 학습, 이상 탐지, 딥러닝, etc
E-mail : bang@sju.ac.kr

유 지 훈(Jihoon Yoo)
2018년 세종대학교 대학원 컴퓨터공학과 (공학석사)
2019년~현재 세종대학교 대학원 박사과정
관심분야 : 분산 처리, 데이터 마이닝, 딥러닝, etc
E-mail : yoojihoon@sju.ac.kr

김 상 수(Sangsoo Kim)
2003년 경북대학교 대학원 컴퓨터공학과 (공학석사)
관심분야 : 사이버전 기술, 위협 헌팅, 사이버 상황인식, etc
E-mail : wisdory@naver.com

◐ 저 자 소 개 ◑

신 동 일(Dongil Shin)
1988년 연세대학교 컴퓨터 과학과 (공학사)
1993년 Washington State University 컴퓨터과학과 (공학석사)
1997년 North Texas University 컴퓨터과학과 (공학박사)
1998년~현재 세종대학교 컴퓨터공학과 교수
관심분야 : 정보 보안, 기계 학습, 데이터 마이닝, 생체 데이터 처리, etc
E-mail : dshin@sejong.ac.kr

신 동 규(Dongkyoo Shin)
1986년 서울대학교 계산통계학과 (공학사)
1992년 Illinois Institute of Technology 컴퓨터과학과(공학석사)
1997년 Texas A&M University 컴퓨터과학과(공학박사)
1998년~현재 세종대학교 컴퓨터공학과 교수
관심분야 : 정보 보안, 기계 학습, 데이터 마이닝, 생체 데이터 처리
E-mail : shindk@sejong.ac.kr