논문 20-45-12-15 The Journal of Korean Institute of Communications and Information Sciences '20-12 Vol.45 No.12
https://doi.org/10.7840/kics.2020.45.12.2114

침입 탐지 데이터의 불균형 문제 개선을 위한 CTGAN 오버 샘플링 기법 도입에 관한 연구

최 윤 희•, 오 경 환°

A Study on the Introduction of CTGAN Oversampling Algorithm to improve Imbalance Problem in Intrusion Detection Data

Yoon-hee Choe•, Kyoung-Whan Oh°

요 약

네트워크 침입 탐지 데이터는 기본적으로 다수의 정상 데이터와 극소수의 공격 데이터로 구성되어 있다. 이러한 데이터 불균형 문제는 소수 데이터의 예측 편향과 이상치로 오판 등의 예측 성능 저하 요인을 야기한다. 불균형 문제를 해소하기 위한 대표적인 방법으로는 SMOTE 알고리즘에 기반한 다양한 소수 데이터 합성 모델이 있다. 그러나, 생성적 적대 신경망(GAN) 모델이 개발된 이후는 이를 활용한 소수 데이터의 합성에 대한 연구가 활발하다. 본 연구에서는 생성적 적대 신경망에 기반한 CTGAN 오버 샘플링 모델을 이용하여 침입 탐지 데이터의 불균형 문제를 해소하고, SMOTE 기반 모델들과 그 성능을 비교한다. 이에 그치지 않고, 유사 공격을 직접 발생 시켜 분류 모델의 실무 적용 가능성까지 확장한다.

키워드 : 침입 탐지, 데이터 불균형, CTGAN 알고리즘, 오버샘플링, 기계학습
Key Words : Intrusion Detection, Imbalanced Data, CTGAN Algorithm, Oversampling, Machine Learning

ABSTRACT

Network intrusion detection data consists essentially of a large number of normal data and very few anomaly data. This data imbalance problem causes predictive performance degradation, such as misjudgment, with predicted bias and anomalies in a small number of data. Typical methods for solving the imbalance problem are various minority data synthesis models based on SMOTE algorithms. However, since the development of the Generative Adversarial Networks(GAN) model, research have been active on the synthesis of minority data using it. In this study, CTGAN oversampling model based on GAN Algorithm is used to solve the imbalance problem of intrusion detection data, and compare its performance with SMOTE-based models. In addition, generate attacks similar to those used in the experiment, extending the practical applicability of the classification model.

※ 본 연구는 서강대학교 정보통신 대학원의 과제로 수행되었음.
• First Author : Sogang graudate School of Information & Technology, yhchoe2010@gmail.com, 학생(석사), 학생회원
° Corresponding Author : Sogang University, Department of Computer Engineering, Artificial Intelligence Lab, kwoh@sogang.ac.kr, 정교수, 정회원
논문번호 : 202007-144-B-RN, Received July 2, 2020; Revised August 18, 2020; Accepted August 25, 2020

Ⅰ. 서 론

침입 탐지 시스템은 네트워크 단(N-IDS) 또는 호스트 단(H-IDS)에 설치되어 공격을 탐지하여 보안 인력에게 알려주는 시스템이다. 침입 탐지 시스템의 탐지 방법은 크게 오용 탐지와 이상 탐지로 나눌 수 있다. 그 중 이상 탐지는 Zero-day 공격까지도 탐지할 수 있다는 장점이 있다. 그러나 기계 학습의 특성상, 정상 행위를 공격으로 분류하는 오탐율(False Alarm Rate)이 높은 단점이 있다.[1] 이로 인해 보안 인력이 수동으로 공격 여부를 판단해야하는 불편함이 있다.

인공 지능 보안 시장이 커지는 것에 비례하여, 인공지능을 이용하여 기존의 관제 시스템을 우회하는 공격도 증가하고 있다. 따라서, 실무에 적용 가능할만큼 낮은 FAR을 가진 기계 학습 방법을 찾는 것이 주요 과제로 꼽히고 있다.[2] 근래는 딥러닝의 활발한 연구로 인해 이를 적용하여 FAR을 낮추려는 다양한 시도가 이루어지고 있다.[3-5] 그 중에서도 생성적 적대 신경망(GAN)을 이용한 연구를 살펴보면, 이상 탐지[6] 및 탐지 회피[7] 단계에서 기존의 학습 모델들에 비해서 좋은 성능을 보임을 알 수 있다. 그러나 예측 모델에 관해 다양한 연구가 이뤄진 반면에 침입 탐지 데이터가 가진 데이터 불균형이라는 근본적인 문제를 전처리 단계에서 해결한 연구는 적은 편이다. 데이터 불균형은 소수 데이터가 많게는 전체의 1:100에서 적게는 1:10000 가량을 차지하는 데이터 셋을 의미한다.[8] 침입 탐지 시스템의 목적은 다수의 정상 패킷 중에서 매우 적은 소수의 공격 패킷을 찾아내는 것이다. 따라서, 항상 데이터 불균형 문제를 안고 있을 수 밖에 없다. 데이터 불균형은 다수 클래스로의 예측 편향과 소수 데이터를 이상치로 오판하여 무시하는 현상을 일으킨다. 이는 모델의 예측 성능을 저하 시키는 요인이 된다.

데이터 불균형 문제를 제거하는데는 다양한 기법이 있으나, 데이터 개수를 조정하는 방법은 언더 샘플링과 오버 샘플링이 있다. 언더 샘플링은 다수 클래스에 속하는 데이터를 소수 클래스 데이터 수 만큼 축소하는 방법으로, 다수 클래스에 속하는 데이터의 정보가 유실되는 한계가 있다. 반면 오버 샘플링은 소수 클래스 데이터를 다수 클래스 데이터 수 만큼 과표본화 하는 기법이다. 대표적인 방법으로 Random Over Sa-mpling[9]과 KNN(K-Nearest Neighbor)[10] 알고리즘에 기반한 SMOTE[11], Borderline-SMOTE[12], ADA-SYN[13]이 있다. 그러나, KNN이 선택된 소수 표본 간의 거리에 기반해서 합성 데이터를 생성하는 알고리즘이기 때문에 다음과 같은 한계가 존재한다.

첫 번째로, 선택된 소수 클래스 데이터의 표본과 가장 가까운 이웃 데이터의 거리가 멀리 있는 경우 다수 클래스의 결정 경계를 침범하게 된다. 즉, 의도하지 않은 분포가 생겨나게 된다. 두 번째로, 이상치에 대해 고려하지 않는다.[14] 또한, 합성 데이터가 중복적으로 생성되어 데이터양은 증가했지만, 정보는 증가하지 못하는 문제도 있다.

이안 굿펠로우가 발표한 딥러닝 기반의 GAN(Generative Adversarial Networks)[15] 알고리즘은 생성기와 판별기 두 개의 부분으로 구성되어, 생성기가 원본과 유사한 합성 데이터를 생성하여 판별기를 속이는 구조를 가진다. GAN의 생성기 부분을 이용하면 원본 이미지와 유사한 합성 이미지를 생성할 수 있다.[16] GAN에 기반한 오버샘플링 기법의 특징은 소수 클래스의 분포를 학습하여 유사한 데이터를 생성한다는 것이다. 또한, 생성한 데이터에 대해 판별기가 원본과의 유사성 여부를 확인하기 때문에 더 양질의 데이터를 생성할 수 있다. 단순히 유클리디안 거리에 기반하여 데이터를 생성해내는 기존 모델의 한계점을 보완 할 수 있다.[17]

이 기능을 구조화된 데이터에서 사용하기 쉽게 고안된 알고리즘이 CTGAN[18]이다. CTGAN은 C-GAN[19]과 TGAN[20]의 장점을 합친 알고리즘으로서 구조화된 데이터에 특화된 오버 샘플링 패키지를 제공한다.

본 논문에서는 침입 탐지 데이터가 가진 클래스 불균형 문제를 CTGAN 패키지를 이용하여 완화하고, 기존의 오버 샘플링 기법인 SMOTE, Borderline-SMOTE, ADASYN과의 성능 차이를 비교한다. 또한 가상 환경에서 공격 패킷을 발생시킴으로써, 학습 모델이 테스트 데이터 셋에만 국한되지 않고 다양한 환경에서 발생되는 동종의 공격에 대해서도 탐지가 가능한지 가늠해볼 수 있다.

Ⅱ. 본 론

2.1 침입 탐지 데이터의 불균형 리샘플링 관련 연구

침입 탐지 데이터는 거래 데이터, 불량품 제거 데이터와 함께 꼽히는 대표적인 불균형 데이터이다. 침입 탐지 데이터 셋은 다수의 정상 데이터와 소수의 침입 데이터로 이루어져 있다. 전처리 단계에서 이러한 데이터의 불균형을 제거하는 작업을 리샘플링(Re-Sampling)이라고 한다. 리샘플링 과정에 의해서 예측

모델의 성능이 변하기[21] 때문에, 적용하려는 리샘플링 모델의 특징을 이해하고 데이터에 적용 시켜야한다.

F. Rahat 등[22] 은 침입 탐지 데이터인 KDD99 셋을 이용하여 전처리 방법에 따른 정확도와 재현율을 비교하였다. 전처리를 거치지 않은 데이터의 분류 성능 오버샘플링으로 K-fold 교차 검증과 Random Oversampling을 채택했고, 특징(Feature) 선택으로는 차원 축소 방법인 주성분 분석(Principal Component Analysis ,PCA)을 적용했다. 전처리를 거친 데이터는 J48, Naïve Bayes, Adaboost, Bagging, Nearest Neighbor, Design. 예측 모델을 이용하여 분류했다. 실험 결과 PCA는 정보 획득량의 증가에 영향을 미쳤지만, 오버 샘플링은 정확도에 큰 영향을 미치지 못하는 결과를 보였다. 그러나 실험에 사용된 Random Oversampling 기법은 소수 클래스 데이터를 다수 클래스 데이터만큼 무작위로 생성하기 때문에 데이터의 중복이 많고 과적합(Overfitting)이 일어날 가능성이 높은 결함을 가지고 있다. D.A. Cieslak 등[23]은 SMOTE, Cluster-SMOTE와 언더 샘플링 리샘플링을 적용한 각각의 데이터를 Ripper 알고리즘을 이용한 성능 평가 연구를 했다. 성능 평가 지표는 ROC-Curve로 C-SMOTE와 SMOTE를 적용한 데이터가 언더 샘플링 전처리를 했을 때보다 좋은 성능을 보임을 나타냈다. Abebe Tesfahun 등[24]은 NSL-KDD 데이터셋을 이용하여 SMOTE와 랜덤포레스트 분류 모델의 성능을 측정했다. 전처리 과정으로 SMOTE, 22개의 특징 선택, 랜덤포레스트 알고리즘을 선택한 모델의 탐지율이 0.963으로 다른 비교 모델들에 비해 최대 0.01만큼 향상된 성능을 보였다. 이는 SMOTE 알고리즘만 사용하였기에, 최적의 성능을 내는 SMOTE 조건에 대해서 탐색한 데에 의의가 있다고 본다.

관련 연구에서 살펴본 바에 의하면, 모든 데이터에서 좋은 성능을 내는 전처리 방법은 없다. 또한 오직 오버 샘플링기법만으로는 큰 성능 향상을 기대할 수 없다. 따라서, 데이터 셋에 알맞은 최적의 성능을 내는 전처리 방법을 탐색하는 것이 필요하다.

2.2 CTGAN 모델

2.2.1 GAN 기반의 CTGAN

GAN(Generative Adversarial Networks)[15]은 딥러닝 기반의 비지도 학습 알고리즘이다. 데이터의 분포를 학습하여 원본과 유사한 가짜 데이터를 생성해내는 생성자(Generator)와 데이터가 원본인지 생성자가 만들어낸 데이터인지를 구분하는 판별자(Discriminator)가 적대적으로 경쟁하는 구조를 가지고 있다. 적대적 구조의 수식은 식 (1)과 같다.

$$\min G \max D [E[\log(D(x))] + E[\log(1-D(G(z)))]] \quad (1)$$

여기서 D(x)는 실제 데이터에 대한 판별기의 분류 값이고, D(G(z))는 생성기가 만들어낸 가짜 데이터에 대한 판별기의 분류 값이다. 수식에서, 판별기 D는 전체 수식 값이 최대 값을 갖도록 학습한다. 즉, D(x)를 1로 예측하고 D(G(z))를 0으로 예측해 내는 것이 목표이다. 반면에 생성기 G는 전체 수식 값이 최소가 되도록 학습한다. 즉, 판별기가 D(G(z))를 1로 예측하는 G(z)를 생성하는 것을 목표로 한다.

수식에서 알 수 있듯이 유사 데이터를 생성하고 원본과 구분하기 어려운 것을 목표로 학습하기 때문에, GAN 모델이 생성하는 합성 데이터가 높은 수준을 가지게 된다. 실제로 이미지 생성에 있어서 원본과 구분하기 어려울 정도의 합성 데이터를 만들어내고 있고, 딥페이크, 자율주행, 이미지 복원 등의 많은 분야에서 연구가 진행되고 있다. 그러나, 이미지 생성에 비해 구조화된 데이터의 생성은 적용하기 어려운 특징이 있다.

첫 번째로, 구조화된 데이터는 컬럼 별로 다양한 타입을(ex. 숫자, 문자, 시간) 갖는다. 두 번째로, 이미지와 달리 비 가우시안 분포를 갖는다. 이미지의 픽셀 값은 가우시안 분포를 따르며, 이는 변환을 이용하여 정규화 될 수 있다. 그러나 구조화된 데이터는 독립적인 각각의 특징(Feature)으로 이루어져 있다. 세 번째로, 컬럼 별로 상이한 분포 모양을 가지고 있기 때문에, 분포의 모드 값을 추정하는 작업이 필요하다. 네 번째로, 생성자가 심각한 불균형이 있는 데이터의 분포를 학습할 때, 소수 클래스의 데이터가 누락되어 학습하지 못할 수 있다. 마지막으로 종속 변수가 원-핫 인코딩으로 표현되어 있다. 생성자는 전체 클래스 범주의 분포를 학습하고 Softmax를 이용한 종속 변수의 값을 생성한다. 그러나 실제 값이 원-핫 인코딩 형태라면, 판별기가 데이터의 유사성을 판단하여 원본과 가짜를 구별하는 것이 아니라, 종속 변수의 원-핫 인코딩 여부를 판단하여 원본과 가짜를 구별하게 된다. [18]

CTGAN[18] 알고리즘은 위와 같이 구조화된 데이터가 GAN 모델에서 겪는 어려움을 개선하였다.

2.2.2 CTGAN 모델 구조

CTGAN은 Conditional-GAN[19]알고리즘과 Tabu

-lar GAN[20] 알고리즘의 혼합형 모델로서, 구조화 데이터 생성에 최적화된 구조를 가지고 있다. CTGAN은 컬럼 별 데이터 분포의 모드 값을 추정하기 위해서, 변분 가우시안 혼합 분포(Variational Gaus -sian Mixture, VGM)를 이용한다. VGM으로 데이터의 모드를 추정하여, 멀티모달 분포와 비가우시안 분포를 고려한 학습을 할 수 있다.

$$P_{ci}(C_{i,j})=\sum^{mi}_{k=1}\mu kN(C_{i,j};\eta_k,\phi_k) \quad (2)$$

Ci,j는 Ci컬럼에 속하는 데이터값, m_i는 VGM으로 추정한 모수, η_k는 k번째 모드값, μ_k와 Φ_k는 각 모드의 가중치와 정규분포 값을 의미한다. 수식 (2)를 이용하여 데이터 값이 모드에 속할 확률 밀도를 구한다. 예를 들어 VGM을 통하여 mi을 3으로 추정했다고 가정할 때, $C_{i,j}$가 3개의 모드 중 어디에 속하는지를 식 (2)를 이용하여 계산하고, 이 확률 밀도 값을 이용하여 가장 유력한 모드를 원-핫 인코딩 값으로 나타낸다. 즉, 세 번째 모드에 속하는 $C_{i,j}$의 원-핫 인코딩 값은 [0,0,1]이 된다. 그리고 수식 (3)에 의해서 $C_{i,j}$값을 세 번째 모드에 맞게 정규화한다.

$$\alpha i,j=Ci,j-\eta 3/4\phi 3 \quad (3)$$

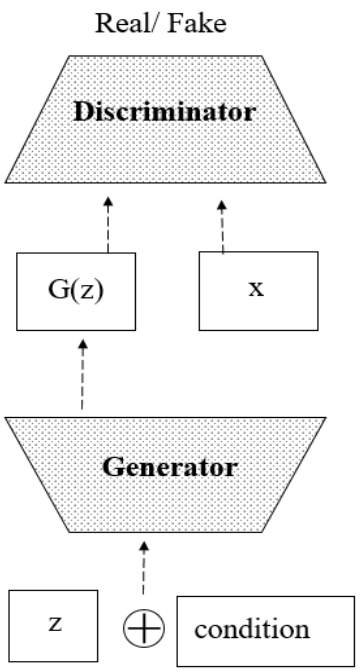

그림 1. CTGAN 신경망 구성
Fig. 1. Model of CTGAN Algorithm

N개의 값에 대해 각 모드에서 정규화된 값과 원-핫 인코딩 벡터 값을 함께 표현하여 최종적으로 r_j 값을 구하는데, 수식은 다음과 같다.

$$r_j=\alpha_{1,j}\oplus\beta_{1,j}\oplus\ldots\oplus\alpha_{Nc,j}\oplus\beta_{Nc,j}\oplus d_{1,j}\oplus d_{Nd,j} \quad (4)$$

α_{Ncj}는 모드에 대해 정규화된 값, β_{ncj} 는 원-핫 인코딩으로 표현된 벡터 값을 의미한다. 수식을 통해서 기존의 GAN 알고리즘에서 구조화된 데이터를 생성할 때 발생했던 문제인 원-핫 인코딩과 멀티 모달 문제를 해결하는 알고리즘을 살펴보았다.

그림 1은 CTGAN 신경망 구성을 도식화하였다. 생성자(Generator)는 잡음 Z와 생성할 데이터의 클래스 정보(Condition)를 가지고 합성 데이터 G(z)를 만들어 낸다. 판별자는 G(z)와 실제 관측치인 x를 가지고 진짜 데이터인지, 합성 데이터인지를 판별하게 된다.

Ⅲ. 실 험

3.1실험 데이터

침입 탐지 연구는 공격의 다양성 등의 이유로 KDD 99[25] 데이터 셋이 주로 사용된다. 그러나 본 연구에서는 유사 공격의 재현을 위해서 CSV-PCAP 변환 프로그램을 지원하는 CICIDS2017[26] 데이터셋을 채택했다. CICIDS2017 데이터는 캐나다의 뉴브런즈윅 대

표 1. CICIDS2017 데이터 공격 별 개수
Table 1. Number of Data by Attacks in CICIDS2017

Category	Attack Name	Data count
-	BENIGN	2,273,097
DoS	DoS Hulk	231,073
	DoS slowloris	5,796
	DoS Slowhttptest	5,499
	DoS GoldenEye	10,293
PortScan	PortScan	158,930
DDoS	DDoS	128,027
	Bot	1,966
Bruteforce	FTP-Patator	7,938
	SSH-Patator	5,897
WebAttack	Brute Force	1,507
	XSS	652
	Sql injection	21
Heartbleed	Heartbleed	11
Infiltration	Infiltration	36
total		2,830,743

학교에서 제공하는 침입 탐지 데이터로, 실험용 네트워크망을 구성하여 다양한 공격을 수행했다.

정상 패킷인 Benign은 2,273,097개로 전체 데이터의 80% 이상을 차지한다. 공격 데이터를 포함하여 총 2,830,743개의 데이터와 15개의 공격명이 있다. 이 중 유사 공격을 묶어서 7개로 범주화하였다.

소수 표본이 상당히 제한적인 경우, 오버 샘플링 알고리즘이 학습할 데이터가 충분하지 않기 때문에 양질의 데이터를 얻기 어렵다. 데이터가 가진 정보가 부족하기 때문에 중복 데이터가 생성될 가능성이 크다. 따라서, 본 실험에서는 합성 데이터의 타당성을 고려하여 공격 데이터의 임계치를 1:10000로 제한했다. 임계치에 따라서 Heartbleed(11개) 공격과 Infiltration 공격(36개) 는 실험에서 제외한다.

또한, 전처리 작업 단계에서 총 79개의 특징(Feature)을 차원 축소 기법인 PCA를 이용하여 8개로 축소 시켰다. 다음으로 스케일러를 적용하여 특징 간 단위 차이로 인한 왜곡을 예방한다. 본 실험에서 사용한 표준화 스케일러는 데이터의 평균과 분산을 이용하여 데이터의 스케일을 조정한다.

3.2 실험 설계

3.2.1 테스트 데이터 환경

문자 타입인 Category를 분류 모델이 분류할 수 있는 숫자 타입으로 변환하는 라벨 인코딩 작업을 통해 Class 0~Class 5로 변환했다. 분류 모델에서 사용할 훈련 데이터와 검증 데이터는 7:3의 비율로 나눠서 준비한다.

테스트 데이터는 가상 머신에서 각각의 공격을 수행하여 확보하였다. 환경의 제약으로 확보하지 못한 DDoS 데이터는 검증 데이터 셋의 10%를 테스트 데이터로 이용한다. 테스트 환경은 웹 서버와 공격 서버로 구성하였다.

웹 서버 환경은 CentOS 6.10 에 Apache2.2.15 / mysql 14.14으로 구축하고, 웹 취약점 공격용 어플리케이션인 DVWA 페이지를 구성하였다.

표 2. 테스트 데이터와 검증 데이터 개수
Table 2. Number of Validation and Test Data

Category	Class	Test Data	Validation Data
BENIGN	0	17	681,347
DoS	1	764	4,150
PortScan	2	3,913	35,088
DDoS	3	1,602	75,504
Bruteforce	4	10,921	47,636
WebAttack	5	128	654

공격 환경은 2019.4-Kali-light 운영 체제를 설치했으며, 공격 도구로는 BruteForce는 Patator, Dos 공격은 Slowloris와 Slowhttptest, PortScan 공격은 Nmap, Bengin(정상패킷)과 WebAttack은 DVWA 웹을 사용하여 재현했다. 패킷 캡처는 tcpdump를 이용하여 수행하고, CIC FlowMeter 프로그램을 이용하여 패킷을 CSV 파일로 변환하였다.

3.2.2 오버샘플링 데이터 사전 평가

전처리가 끝난 데이터는 SMOTE, Borderline-SMOTE, ADASYN, CTGAN 오버샘플링 알고리즘을 적용한다. 분류 모델로 예측하기 전, 각 알고리즘별 합성 데이터의 중복 비율을 비교한다. 데이터 높은 중복은, 소수 데이터의 개수는 증가하였지만 데이터가 가진 정보는 증가하지 못함을 의미한다. 따라서, 성능 향상을 기대하기 어렵다. 또한, 합성 데이터의 중복은 과적합의 요인이 된다. 따라서, 오버샘플링으로 인한 중복 데이터가 얼마나 발생했는지를 확인하여 샘플링별 품질을 판단할 수 있다.

표 3 은 오버 샘플링에 따른 중복 데이터 수를 나타내었다. Train은 기존 학습 데이터를 의미한다. CTGAN은 Train이 가지고 있던 191,851개의 중복 데이터 외에는 새로운 중복 데이터가 0건 발생하였다. ADASYN, Borderline-SMOTE, SMOTE 순으로 중복 데이터가 많이 생성되었다. SMOTE는 전체 데이터 중 약 14.6%가 중복 데이터로, 가장 저조한 데이터 합성 성능을 보였다.

표 3. 오버샘플링 별 중복 데이터 수
Table 3. Number of Duplication Data by Oversampling Algorithm

Oversampling Type	Counts of Duplication	Ratio of Duplication	Rank
Train	191,851	0.096927978	-
SMOTE	1,394,347	0.146175434	1
Borderline-SMOTE	870,333	0.091240777	2
ADASYN	728,226	0.076347791	3
CTGAN	191,851	0.020112571	4

3.2.3 분류 모델 및 성능 평가 척도

데이터 성능을 검증할 분류 모델로 RandomForest[27]와 LightGBM[28]을 사용한다. 실험에 사용한 RandomForest 매개변수는 max_depth: 30, min_sampl es_split:100 그리고 n_estimate: 50, 100일 때를 각각 실험에서는 RF1, RF2라고 지칭한다.

실험에 사용한 LightGBM 매개변수는 max_depth:

30, learning_rate:0.01, object: 'multi:softmax' 그리고 n_estimate:50, 100일 때를 각각 실험에서는 Lig-htGBM1, LightGBM2 라고 지칭한다. 분류 모델의 성능 평가 척도는 F1스코어로, 불균형 데이터 셋의 대표적인 평가지표이다. F1 점수는 재현율(Recall)과 정밀도(Precision)의 조화 평균을 이용하여 구한다. 재현율과 정밀도는 특정 클래스의 올바른 분류에 대해 나타내므로 값이 클수록 좋은 분류 모델이라고 할 수 있다. 즉, F1 점수가 높으면 모델의 성능이 좋다고 평가한다.

3.3 실험 결과

3.3.1 F1 스코어

CTGAN의 F1 스코어가 0.377로 가장 높았다. 공통적으로 모든 샘플링 데이터에서 LightGBM 모델로 예측 했을 때 가장 우수한 성능이 관측되었다. 테스트 데이터 셋의 성능이 전체적으로 저조한 반면, 검증 데이터셋(Validation)은 대부분 0.8 이상 또는 그에 가까운 성능을 보였다. 전처리 작업만 수행한 불균형 데이터(Imbalanced Data)는 오버샘플링 데이터와 비교하여 성능에 큰 차이가 없었다. 이는 선행 연구에서 SMOTE를 이용했을 때 성능에 큰 변화가 없었던 것과 동일한 결과임을 알 수 있다. 그러나 CTGAN 데이터와 불균형 데이터를 비교해보면, 테스트 데이터 셋에서 0.1 만큼의 성능 향상을 보인다. CTGAN 데이터가 다른 오버샘플링 모델에 비해서, 학습 데이터에 국한되지 않는 좀 더 강건한 모델임을 확인 할 수 있다.

표 4. 오버샘플링 별 F1 스코어
Table 4. F1 score by Oversampling Algorithm

Over-sampling	Classifier	F1-score	
		Validation	Test
Imbalanced Data	RF1	0.997	0.222
	RF2	0.997	0.222
	LightGBM1	0.806	0.0009
	LigthGBM2	0.978	0.275
SMOTE	RF1	0.998	0.224
	RF2	0.998	0.224
	LightGBM1	0.954	0.293
	LigthGBM2	0.959	0.294
Borderline-SMOTE	RF1	0.997	0.225
	RF2	0.996	0.225
	LightGBM1	0.912	0.273
	LightGBM2	0.914	0.283
ADASYN	RF1	0.995	0.225
	RF2	0.995	0.225
	LightGBM1	0.885	0.293
	LightGBM2	0.884	0.295
CTGAN	RF1	0.990	0.297
	RF2	0.991	0.297
	LightGBM1	0.788	0.377
	LightGBM2	0.815	0.369

3.3.2 ROC-Curve

ROC-Curve는 얼마나 해당 클래스로 정확하게 분류 했는지를 시각화 할 수 있는 평가지표이다. 전체적인 분류 성능을 볼 수 있으며, 클래스 별 분류 성능을 판단할 수 있다. 다음은 각 오버샘플링 알고리즘 별로 Test 데이터 셋에 대한 ROC-curve를 나타낸 그래프이다.

범례의 micro-average는 조화 평균을 나타내고, macro-average는 일반 평균이다. 숫자가 클수록 재현율(Recall)이 높다는 것이고, 이는 클래스를 정확하게 분류했음을 의미한다. 범례의 각 숫자는 AUC 스코어를 나타내고 1에 가까울수록 분류 성능이 좋음을 의미한다.

임계치는 그래프 중앙의 검은 점선이며 0.5로 지정하였다. 만약 0.5 미만의 AUC 스코어를 보이면, 실제 활용하기에는 오버샘플링의 성능이 부적절한 것으로 판단하였다.

그림 2부터 5는 Test 데이터에 대해서 각 오버 샘플링 데이터 별 AUC 스코어를 보여준다. SMOTE는 클래스 0,1,5에 대해서 임계치 0.5 미만의 AUC스코어를 기록했다. Borderline-SMOTE는 클래스 1, 5에 대해서 임계치 보다 낮은 분류 성능을 보였다. CTGAN은 1, 5 클래스에서 임계치인 0.5 미만으로 나타났다. ADASYN은 클래스 4,5에 대해서 임계치보

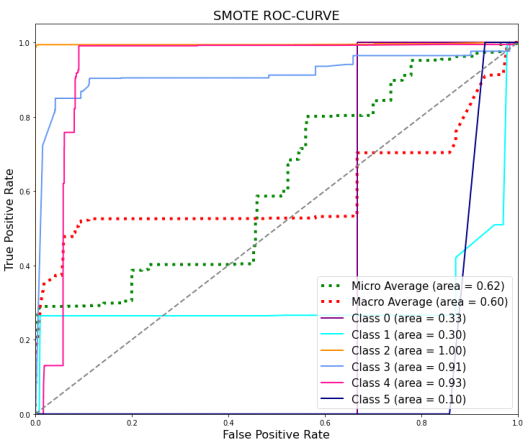

그림 2. SMOTE 알고리즘의 ROC-CURVE
Fig. 2. ROC-CURVE of SMOTE

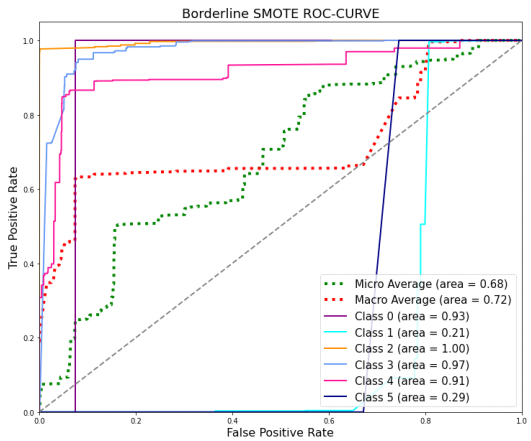

그림 3. Borderline-SMOTE 알고리즘의 ROC-CURVE
Fig. 3. ROC-CURVE of Borderline-SMOTE

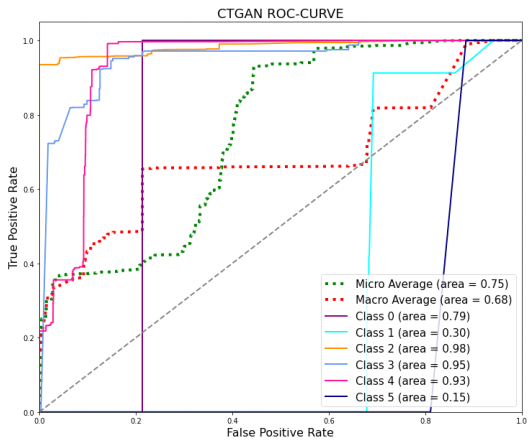

그림 4. CTAN 알고리즘의 ROC-CURVE
Fig. 4. ROC-CURVE of CTGAN

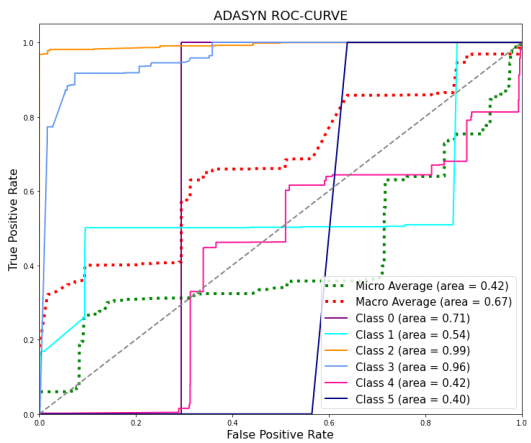

그림 5. ADASYN 알고리즘의 ROC-CURVE
Fig. 5. ROC-CURVE of ADASYN

다 낮은 분류 점수가 관측되었지만, 0.4대의 AUC 스코어로 다른 오버 샘플링 모델들보다는 높은 점수를 보였다. 공통적으로 5번 클래스에 대해서 임계치 미만의 성능을 보였으며, 1번 클래스도 마찬가지로 잘 예측하지 못하는 것으로 확인됐다. ADASYN을 제외한 나머지 오버샘플링의 조화 평균(Micro Average) AUC 점수가 0.5 이상으로 나타나기 때문에 F1 스코어의 임계치를 조정한다면 좀 더 높은 점수를 얻을 수도 있겠지만, 이미 일부 클래스의 분류 성능이 0.5에도 미치지 못하기 때문에 여전히 실무에 적용하기 어렵다고 판단된다.

Ⅳ. 결 론

본 실험에서는 침입 탐지 데이터의 불균형 문제를 해결하기 위하여 CTGAN 오버샘플링 알고리즘을 다루고, 성능을 검증해보았다. CTGAN은 패키지를 통한 사용으로 활용이 쉽고, 다른 KNN 기반 알고리즘에 비해 중복이 한 건도 없는 양질의 합성 데이터를 만드는 장점을 가지고 있는 것을 확인할 수 있었다. 분류 모델로 채택한 랜덤포레스트와 LightGBM은 성능 비교 결과, LightGBM이 더 높은 분류 성능과 빠른 예측 속도를 보였다.

평가 단계에서는 학습 모델을 다른 환경에서도 적용 할 수 있는지 확인할 수 있도록, 학습에 쓰인 공격을 다른 환경에서 구현하여 테스트 데이터로 활용하였다. 그 결과, 성능 평가 지표인 F1 스코어에서 최고 성능이 0.377로 저조한 성능을 나타냈다. 해당 성능은 LightGBM 모델에서 CTGAN 오버샘플링 데이터를 학습 시켰을 때 관측 되었다. 이는 실무에 적용하기에는 다소 어려운 성능이다.

본 실험의 결과가 동일한 오버샘플링 기법 또는 분류 모델을 채택한 타 연구들에 비해 저조한 성능을 보이는 이유는 학습 데이터와 다른 네트워크 환경에서 테스트 데이터를 생성했기 때문이다. 같은 공격이더라도 공격 도구, 공격자의 환경 또는 대상 시스템의 종류, 망 구성 등 여러 가지 요인에 의해서 패킷 지속 시간, 전송한 바이트 크기 등이 다를 수 밖에 없다. 이 점이 기계 학습 알고리즘을 실무에 바로 적용 시키기 어려운, 높은 오탐율의 요인이 된다.

본 실험에서는 공격을 재현해봄으로서, 데이터 불균형을 해소하는 것만으로는, 학습 데이터 셋과 다른 환경에서 생성된 데이터 셋에서의 성능을 기대하기 어려움을 다시 한번 확인하는 계기가 되었다.

연구가 시사하는 바는 우선, CTGAN 오버샘플링

이 실험자의 특별한 개입이 없이도 쉽게 양질의 합성 데이터 생성이 가능하다는 것이다. 두번째로는 학습용 데이터 셋과 동일한 공격을 구현하여 테스트 데이터로 활용함으로서, 실무에 적용 가능성을 판단할 수 있었다는 것이다. 결과적으로 침입 탐지 데이터는 네트워크 환경에 의해 특징(Feature)값이 달라지기 때문에, 우수한 성능을 보인 실험 모델이라도 실무에 적용 여부는 고려해야 할 문제라는 것을 알 수 있었다.

본 논문에서는 랜덤포레스트, LightGBM 두 가지의 분류 모델만 활용했지만, GAN, ANN 등의 신경망 알고리즘을 다양하게 사용하여 최고의 성능을 낼 수 있는 모델을 찾을 수 있을 것이다. 더 나아가서, 같은 공격이라면 테스트 환경이 달라져도 동일하게 분류해 낼 수 있는 강건한 모델에 관한 연구를 통해, 실무에 쉽게 적용 가능한 후속 연구를 기대한다.

References

[1] Institute of Financial Security Research, "*Trends in machine learning-based abnormal transaction detection systems*(2017)," Retrieved Jun. 28, 2020, from https://www.fsec.or.kr/common/proc/fsec/bbs/42/fileDownLoad/1269.do

[2] K. Gook, B. Gong, "Development trend of security technology using artificial intelligence," *Weekly ICT Trends by IITP*, vol. 1913, no. 1, pp. 2-15, Sep. 2019.

[3] S. Supriya and T. Samrat, "A review on deep learning method for intrusion detection in network security," *IEEE ICIMIA, pp. 173-177, Bangalore, India, March 2020.*

[4] W. Chen, F. Mei, F. Kong, G. Yuan, and B. Li, "A novel unsupervised anomaly detection approach," *IEEE 3rd Int. Conf. Big Data Security on Cloud, pp. 69-73, Beijing, China, May 2017.*

[5] R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: A survey," *arXiv:1901.03407v2*, Jan. 2019.

[6] H. Chen and L. Jiang, "Efficient GAN-based method for cyber-intrusion detection," *arXiv : 1904.02426*, Jul. 2019.

[7] M. A. Ayub, W. A. Johnson, D. A. Talbert, and A. Siraj, "Model evasion attack on intrusion detection systems using adversarial machine learning," *54th Annual Conf. on Information Science and Systems (CISS), pp. 1-6, Princeton, NJ, USA, March 2020.*

[8] H. He and E. A. Garcia, "Learning from imbalanced data". *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263-1284, Sep. 2009.

[9] C. X. Ling and C. Li, "Data mining for direct marketing: Problems and solutions", *KDD'98: Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*, pp73-79, Aug 1998.

[10] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory,* vol. 13, no. 1, pp. 21-27, Jan. 1967.

[11] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artficial Intell. Res.,* vol. 16, pp. 321-357, Jun. 2002.

[12] H. Han, W. Y. Wang, and B. H. Mao, "Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning," *LNCS,* vol. 3644, pp. 878-887, 2005.

[13] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," *2008 IEEE IJCNN*, pp. 1322-1328, Hong Kong, China, Jun. 2008.

[14] W. Chen, F. Mei, F. Kong, G. Yuan, and B. Li, "A novel unsupervised anomaly detection approach," *IEEE 3rd Int. Conf. Big Data Secur. Cloud,* pp. 1281-1286, 2017.

[15] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *NIPS*, 2014.

[16] A. Ali-Gombe, E. Elyan, and C. Jayne, "Multiple fake classes GAN for data augmentation in face image dataset," *Int. Joint Conf. Neural Netw.*, Budapest, Hungary, 2019.

[17] G. Douzas and F. Bacao, "Effective data generation for imbalanced learning using conditional generative adversarial networks," *Expert Syst. with Appl.,* vol. 91, pp. 464-471,

Jan. 2018.

[18] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling tabular data using conditional GAN," *NeurIPS*, 2019.

[19] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv:1411.1784v1*, 2014.

[20] K. V. Lei Xu, "Synthesizing tabular data using generative adversarial networks," *arXiv:1811.11264*, 2018.

[21] K. Lee, J. Lim, K. Bok, and J. Yoo, "Handling method of imbalance data for machine learning : Focused on sampling," *The J. Korea Contents Assoc.*, vol. 19, no. 11, pp. 567-577, 2019.

[22] F. Rahat and S. N. Ahsan, "Comparative study of machine learning techniques for pre-processing of network intrusion data," *ICOSST*, Lahore, Pakistan, Dec. 2015.

[23] D. A. Cieslak, N. V. Chawla, and A. Striegel, "Combating imbalance in network intrusion datasets," *IEEE Int. Conf. Granular Comput.*, pp. 732-737, May 2006.

[24] A. Tesfahun and D. L. Bhaskari, "Intrusion detection using random forests classifier with SMOTE and feature reduction," *Int. Conf. Cloud & Ubiquitous Comput. & Emerging Technol.*, pp. 127-132, Nov. 2013.

[25] *KDD Cup 1999 Data*, Retrieved Jun. 28, 2020, from http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[26] *Intrusion Detection Evaluation Dataset*(CIC-IDS2017), Retrieved Jun. 28, 2020, from https://www.unb.ca/cic/datasets/ids-2017.html

[27] L. Breiman, "Random Forests," *Mach. Learning,* vol. 45, pp. 5-32, Oct. 2001.

[28] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T. Y. Liu1, "LightGBM: A highly efficient gradient boosting," *31st Conf. NIPS, Long Beach, CA, USA, 2017.*

최 윤 희 (Yoon-hee Choe)

2013년 2월 :서일대학교 컴퓨터전자과 졸업

2012년 12월~2016년 6월 : 이너버스 근무

2016년 11월~2019년 2월 : 구우정보기술 근무

2020년 8월 : 서강대학교 정보통신대학원 졸업

<관심분야> 정보보안, 딥러닝, 보안 로그 분석

[ORCID:0000-0003-3479-8690]

오 경 환 (Kyung-Whan Oh)

1978년 2월 : 서강대학교 수학과 졸업

1985년 5월 : Florida State University, Department of Computer Science, 석사

1988년 12월: Florida State University, Department of Computer Science, 박사

1989년 3월-현재 : 서강대학교 컴퓨터공학과 교수

<관심분야> 인공지능, 지능형 에이전트, 딥러닝