Journal of the Korea Academia-Industrial cooperation Society
Vol. 24, No. 3 pp. 124-130, 2023

https://doi.org/10.5762/KAIS.2023.24.3.124
ISSN 1975-4701 / eISSN 2288-4688

훈련데이터의 속성 재배열 방법에 따른 CNN 기반의 IDS 성능 비교

박휘랑, 조영호*
국방대학교 국방관리대학원 국방과학학과(컴퓨터공학/사이버전협동전공)

Performance Comparison of CNN-based IDS through Various Feature Rearrangement of Training Dataset

Hweerang Park, Youngho Cho*
Department of Defense Science, Korea National Defense University

요 약 CNN(Convolutional Neural Networks)은 이미지의 특징을 추출하는 방법을 사용하여 이미지 분류에 좋은 성능을 보이는 딥러닝 모델이다. CNN은 사이버보안 분야의 IDS(Intrusion Detection System) 연구에서도 활발히 활용되고 있다. CNN 기반의 IDS(CNN IDS) 구축을 위해서는 문자열로 구성된 IDS 데이터를 이미지 형태로 변환하는 전처리 과정을 거쳐야 한다. 기존의 CNN IDS 연구들을 살펴보면 훈련데이터의 속성(Feature) 순서의 변경 없이 그대로 배치하여 훈련하는 방식만을 사용했으나, 다른 CNN 연구들을 살펴보면 전처리 과정에서 다양한 속성 재배열 방법을 사용하고 있다. 따라서, 본 연구에서는 CNN에서 활용되고 있는 다양한 속성 재배열 방법들을 CNN IDS의 훈련데이터에 적용하여 CNN IDS의 성능 변화에 대한 연구를 수행하였다. 실험결과, NSL-KDD 데이터세트에 랭크넷(Ranknet) 속성 재배열 방법을 적용했을 때 CNN IDS의 정확도가 기본 속성 배열 방식에 비해 최대 3.7%P의 성능이 향상됨을 확인하였고 이를 통해서 기본 배치 방식은 정확성 측면에서 최적이 아님을 확인하였다.

Abstract CNN (Convolutional Neural Networks) is a deep learning model that performs well in image classification using a method of extracting features of images. CNN is used widely in IDS (Intrusion Detection System) research. The IDS dataset composed of strings must be converted into images in the preprocessing process to construct a CNN-based IDS (CNN IDS). According to the survey, existing CNN IDS studies do not change the order of the features of the training data, while in other CNN research fields, various feature rearrangement methods are used in the preprocessing process. Therefore, this study compared a CNN-IDS by applying multiple feature rearrangement methods to the training data of CNN IDS. According to the experimental results, when the RankNet feature rearrangement method is applied to the NSL-KDD dataset, the accuracy of CNN IDS was improved by up to 3.7%P compared to the basic feature arrange method. Therefore, the basic arrangement method in CNN IDS is not the best choice in terms of classification accuracy.

Keywords : Deep Learning, CNN, IDS, Feature Rearrangement, NSL-KDD, UNSW-NB15, CIC-IDS2017

*Corresponding Author : Youngho Cho(Department of Defense Science, Korea National Defense University)
email: youngho@kndu.ac.kr
Received January 3, 2023 Revised February 6, 2023
Accepted March 3, 2023 Published March 31, 2023

1. 서론

CNN(Convolution Neural Networks)은 최초 제안된 이후 자연어처리, 컴퓨터 비전 분야 등에서 활발히 활용되고 있다[1]. 또한, CNN은 IDS(Intrusion Detection System) 분야에도 활용되어 높은 탐지 성능을 보여주고 있다[2-4]. CNN IDS에 사용되는 IDS 훈련데이터는 네트워크 트래픽 로그를 수집한 것으로 문자열을 기반으로 구축되어 있다. 그러므로 CNN IDS를 구축할 때 문자열을 이미지로 변환하는 선행 과정이 반드시 필요하다. 이 과정은 전처리 과정에서 수행되며 어떤 이미지 변환 방법을 사용하느냐에 따라 훈련된 모델의 성능은 달라지게 된다.

일반적인 CNN 연구에서는 전처리 단계에서 다양한 속성 재배열 방법들이 사용되고 있지만[5,6], 기존의 CNN IDS의 연구에서는 특별한 이유 없이 훈련데이터의 속성 재배열 처리 없이 훈련데이터를 학습하고 있다[2-4].

본 연구에서는 IDS 훈련데이터에 대한 다양한 속성 재배열 방법들을 적용하여 CNN IDS의 성능 변화를 확인해보고자 한다. 실험결과, NSL-KDD 데이터세트에 랭크넷 속성 재배열 방법을 적용했을 때 CNN IDS의 정확도가 기본 속성 배열 방식에 비해 최대 3.7%P의 성능이 향상됨을 확인하였고 이를 통해서 기본 배치 방식은 정확성 측면에서 최적이 아님을 확인하였다.

이후 논문 구성은 다음과 같다. 2장에서 본 연구와 관련된 배경 지식과 기존 연구들을 소개하고, 다른 CNN 연구 분야에서 적용했던 속성 재배열 방법들을 기술한다. 3장에서는 속성 재배열 방법에 따른 CNN IDS의 정확도를 비교실험을 수행하고, 4장에서 향후 연구계획과 함께 결론을 맺는다.

2. 배경지식 및 관련연구

2.1 CNN의 특징과 훈련과정

사람의 눈은 사물의 특징들을 인식하여 분류하지만, CNN은 다수의 커널들(Kernels)을 사용하여 가중치를 수치화하고 생성된 특징 지도(Feature maps)를 사용하여 이미지를 분류한다. CNN은 합성곱과 풀링 방법으로 이미지의 특징들을 찾아내고 그 특징들을 이용하여 이미지를 분류한다. 이러한 방법은 사람이 사물을 인식하는 것과 유사하다[7].

일반적인 CNN의 훈련과정은 다음의 세 단계로 진행된다. 우선, 데이터를 수집하고 속성을 생성한다. 그다음은 전처리과정으로, 데이터를 특정 방법으로 가공하여 이미지로 변환한다. 마지막으로, 데이터를 훈련데이터, 검증데이터, 테스트데이터로 구분하고 이를 활용하여 훈련하고 평가한다[1].

2.2 IDS 데이터세트의 종류와 구성

IDS 연구에 사용되는 데이터세트는 대표적으로 NSL-KDD, UNSW-NB15, CIC-IDS2017 등이 있다[8-10]. Table 1에서 각 데이터세트의 특징을 비교하였다.

Table 1. 3 IDS Dataset Features

		NSL-KDD	UNSW-NB15	CIC-IDS2017
Features		43	45	47
Records	Train Data	125,973	82,332	1,042,557 (Split 6:4)
	Test Data	22,544	175,341	
File Size		21.5 MB	45.4 MB	365 MB

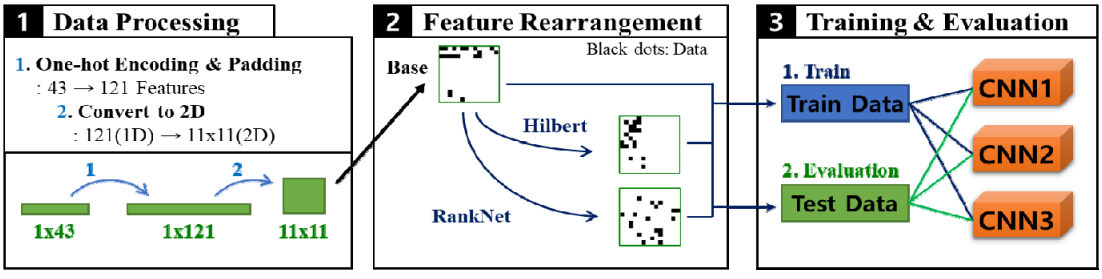


Fig. 1. Feature Order Rearrangement Experiment Procedure(NSL-KDD Dataset case)
(1) Data Processing: Data conversion for 2-dimensional arrays. (2) Feature Rearrangement: Generate 3 types of experimental data. (3) Training and Evaluation: Training and evaluation by dividing data into training data and test data.

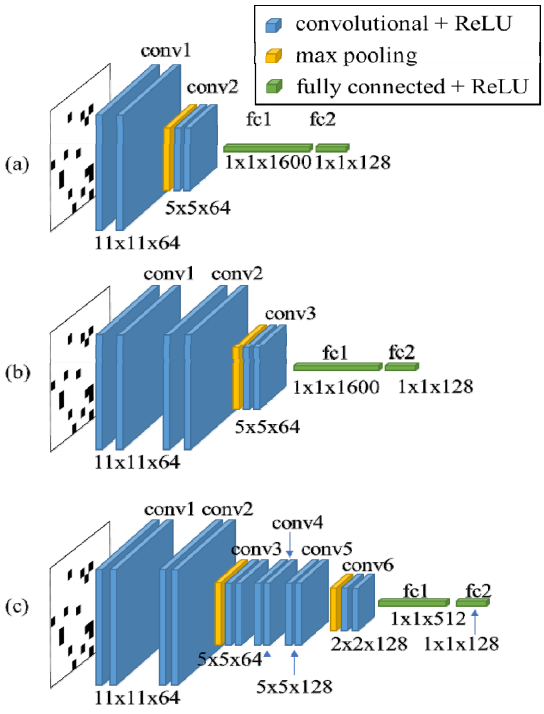


Fig. 2. Experiment CNN IDS Models(NSL-KDD Dataset)
(a) CNN1: 1 convolutional layer with 205k parameters.
(b) CNN2: 2 convolutional layer with 242k parameters.
(c) CNN3: 4 convolutional layer with 324k parameters.

2.3 CNN 전처리 과정과 속성 재배열 기법

SR. Vinayakumar 등은 NSL-KDD 데이터세트을 사용하여 여러 가지의 CNN IDS 모델들을 제안했다[2,8]. Zhang 등은 NSL-KDD 데이터세트을 사용하였다[3,8]. 전처리 과정에서 41개의 속성을 11x11의 2차원 이미지 배열로 변환하여 훈련했다. Roberto Blanco 등은 UNSW-NB15 데이터세트에 유전 알고리즘(Genetic Algorithm)을 적용하여 모델의 성능을 높였다[4,9]. 위 연구들은 데이터 속성의 순서를 재배열하지 않고 그대로 사용했다.

다른 CNN 분야에서 속성 재배열 방법에 대해 여러 가지 연구가 진행되었다. Ren 등은 힐베르트 곡선(Hilbert Curve)에 따라 악성코드의 데이터를 곡선을 따라 배치하여 훈련데이터를 생성했다[5]. Yuyu Luo 등은 자율주행 분야에서 운전자의 시각정보 등을 문자열 데이터로 구축하고 시각화했으며, 시각화에는 데이터를 순위를 매기는 방법(LTR, Learning To Rank)과 독자적인 방법을 사용했다[6].

3. 속성 재배열 방법별 성능 비교 실험

본 논문의 실험은 Fig. 1과 같은 순서로 진행된다. ① 데이터 가공(Data Processing), ② 속성 재배열(Feature Rearrangement), ③ 훈련 및 테스트(Training and Evaluation)의 세 단계 순서다.

① 단계에서 IDS 데이터세트 3개(NSL-KDD, UNSW-NB15, CIC-IDS2017)를 2차원 데이터로 생성한다[8-10]. ② 단계에서 속성 재배열방법 3개(기본속성배열, 힐베르트 곡선, 랭크넷)를 사용한다. ③ 단계에서 R. Vinayakumar이 제안한 CNN 모델 3개(CNN1, CNN2, CNN3)를 사용하여 총 27개의 사례를 비교했다[2]. 그리고 지표로는 정확도, F1 score를 사용했다. 훈련데이터와 테스트데이터를 구분하여 지표를 측정했다.

3.1 데이터 가공

먼저, ① 데이터 가공(Data Processing) 단계는 2장에서 언급한 NSL-KDD, UNSW-NB15, CIC-IDS2017 데이터세트를 사용하고, 문자열 데이터를 숫자로 변환하는 전처리 과정을 거친다[8-10].

데이터세트는 일부 문자열로 구성되어 있으므로, 문자열을 숫자로 바꾼다. 이를 위해 문자열 속성은 원-핫 인코딩(one-hot encoding)을 적용하여 속성 수를 늘리고, 2차원 크기를 만들기 위해, 제곱수에서 부족한 칸은 '0'값을 채우는 패딩 과정을 거쳤다. NSL-KDD 데이터세트(이하 NSL-KDD)는 Zhang 등이 사용한 방법을 그대로 사용하여 속성을 121개(11×11)로 늘렸다[3,8]. UNSW-NB15 데이터세트(이하 UNSW-NB15)는 43개 속성 중에서 3개의 문자열 속성을 원-핫 인코딩하여 각각 131, 13, 9개 속성을 추가하고, 패딩으로 6개 속성을 추가하여, 총 196(14×14)개 속성으로 구성했다[9]. CIC-IDS2017 데이터세트(이하 CIC-IDS2017)는 패딩으로 3개의 속성을 추가하여 81(9×9)개의 속성을 구성했다[10]. 그리고 CIC-IDS2017은 4개의 라벨이 존재하여 다중분류(multiclass classification)로 구성했다[10]. 테스트데이터도 훈련데이터와 같은 기준으로 전처리과정을 거친다. 마지막은 표준화(Normalization)과정으로, 숫자 데이터를 0에서 1 사이의 값으로 변환하였다.

3.2 속성 재배열

② 속성 재배열(Feature Rearrangement) 단계는 기본방법에 두 가지 속성 재배열 방법을 추가로 적용하는

단계이다. 속성을 재배열하는 방법은 공간채움 곡선 중에서 힐베르트 곡선 모양대로 재배열하는 방법과 LTR 알고리즘 중 랭크넷(RankNet)으로 속성의 순위를 매기는 방법을 적용했다[11]. 힐베르트 곡선은 디귿(ㄷ)자 모양의 조각(4칸)을 연속적으로 이어붙여서 공간을 채우는 방법이다. 랭크넷은 마이크로소프트 빙 검색 서비스에서도 사용되는 방법이며, 이진분류로 2개씩 데이터를 비교하여 데이터의 순서를 정렬하는 방법이다[11]. Fig. 1에서 녹색 네모 그림은 데이터가 있는 값은 검은색으로 표시하여 재배열 방법별 데이터의 분포를 나타냈다.

이 단계는 기존 CNN 연구에는 적용했지만 CNN IDS 분야에는 적용되지 않았던 속성 재배열 방법들을 적용하는 단계이다. 우리는 본 논문의 실험에서 이 속성 재배열 단계를 적용해서, 속성 재배치 여부에 따른 성능을 비교해보고자 한다.

3.3 훈련 및 테스트

③ 훈련 및 테스트(Training and Evaluation) 단계는 2단계에서 생성한 데이터를 사용하여 세 가지 CNN 모델을 훈련하고, 훈련된 모델의 성능을 평가하는 단계이다. 실험에 사용하는 CNN IDS 모델은 Fig. 2와 같이 R. Vinayakumar이 제안한 모델인 CNN 1 layer(이하 CNN1), CNN 2 layer(이하 CNN2), CNN 3 layer(이하 CNN3)를 사용했다[2]. 121(11×11)개 속성의 데이터를 입력했을 때 파라미터 수는 각각 약 20만 개, 24만 개, 32만 개다. CIC-IDS2017은 다중분류이므로 출력계층을 1개에서 4개로 변경하고 활성화 함수는 시그모이드(sigmoid)에서 소프트맥스(softmax)로 변경하였다.

속성의 순서 재배열를 위해 세 가지 방법(기본 속성배열, 힐베르트 곡선, 랭크넷)을 사용한다.

최근 연구에서는 딥러닝을 다양한 환경에 적용하기 위해 모델의 정확도 등은 크게 떨어트리지 않으면서 파라미터 수를 줄이는 연구가 진행되고 있다[12]. 파라미터 수는 훈련 및 예측(Predict)에 소요시간에 영향이 있으므로 파라미터 수를 줄이면 이 소요시간도 줄어든다. 이와 같이 파라미터 수에 따른 실험도 필요하다고 보고, 본 논문에서는 파라미터 수가 다른 세 개의 모델으로 실험했다.

3.4 실험 결과

훈련에 소요되는 시간으로, NSL-KDD은 총 46분이 소요되었고, 모델 하나당 평균 약 5분이 소요됐다. UNSW-NB15은 총 1시간 57분이 소요되었고, 모델 하나당 평균 17분이 소요됐다.

실험 결과는 Table 2, Fig. 3과 같다. Table 2는 훈련데이터와 테스트데이터의 결과값을 비교하기 위해 지표로서 정확도(accuracy), F1스코어(F1 score)를 사용하여 비교했다. F1스코어은 분류 성능을 측정하고 정확도와 비교하기 위해 사용했다. 오차를 고려하여 결과 값이 1.0%P 보다 증가한 경우만 파란색으로 표시했다. 테스트 성능은 순차적으로 증가하지 않기 때문에 단순 비교가 어려우므로, 최댓값을 기준으로 비교했다.

Table 2의 훈련데이터의 정확도 항목을 보면, NSL-KDD는 큰 차이가 없거나 오히려 조금 낮았다. UNSW-NB15에서는 CNN1에서 랭크넷이 2.3%P(퍼센트 포인트) 높은 성능을 보여줬다. F1스코어는 CNN2에서 1.5%P 높았다. CIC-IDS2017는 성능이 기본적으로 모두 90%이상의 높은 성능을 보여서 1.0%P 미만의 성능향상만 보여줬다. 훈련데이터에서는 CNN 파라미터 수에 따른 특별한 경향성을 보여주지는 않았다.

다음으로 테스트데이터 항목을 보면, NSL-KDD는 CNN1에서 랭크넷 방법이 최댓값은 3.7%P 높았다. UNSW-NB15는 CNN3에서 랭크넷 방법이 최댓값은 1.4%P 높았다. CIC-IDS2017에서는 데이터가 훈련데이터와 마찬가지로 기본 성능이 높아 1.0%P 이상의 성능향상은 없었다.

3.5 실험결과 분석

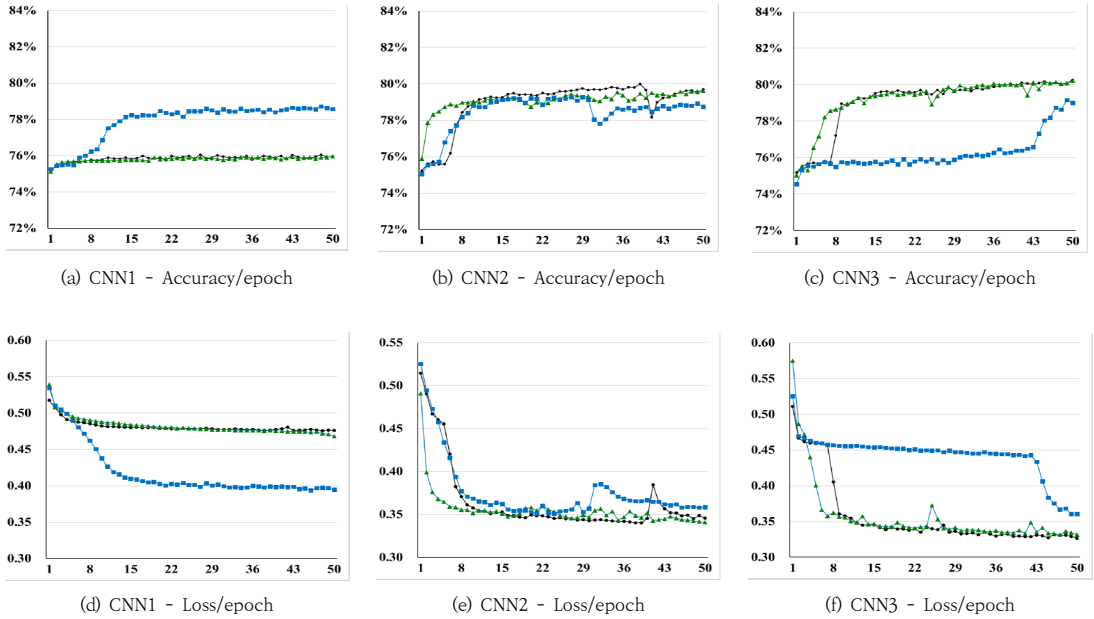
첫째, 데이터세트별로 비교했을 때 테스트데이터는 데이터세트별로 차이를 보여주었다. NSL-KDD은 파라미터 수가 적어질수록 속성 재배열에 대한 효과가 컸고, 상대적으로 UNSW-NB15는 파라미터가 많은 모델에서 좋은 효과를 보여주었다.

둘째, 방법별로 비교했을 때, 전체적으로 힐베르트 방법보다는 속성별 연관성을 고려한 랭크넷이 더 높은 성능향상을 보여줬다. NSL-KDD의 랭크넷의 실험결과를 살펴보면, 파라미터가 특정 수 이하인 모델에서 효과를 나타내는 것을 보였다. 이를 보아서는 규칙 없이 배열하는 것보다는 랭크넷과 같이 데이터 간에 연관성을 갖는 데이터를 근접하게 배치하면 성능이 높아짐을 알 수 있었다.

끝으로, 훈련 단계의 정확도와 테스트데이터를 통한 평가 단계의 정확도의 차이를 작게 만드는 모델의 일반화(generalization) 관점에서 분석하고자 한다. NSL-KDD 데이터세트로 훈련한 경우, 속성 재배열 방법이 훈련 단계에서는 큰 차이를 보이지 못하지만, 테스트데이터를 활용한 평가 단계에서는 차이를 보였다. 훈련 단계는 기

Table 2. Experimental results: Comparison of accuracy by model and image conversion method (Blue: 1.0%P larger value than Base)

Metrics	Dataset Name	Rearrangement Method	Train Data(Maximun)			Test Data(Maximun)		
			CNN1	CNN2	CNN3	CNN1	CNN2	CNN3
Accuracy	NSL-KDD	Base	99.2 %	99.7 %	99.8 %	79.2 %	79.5 %	80.2 %
		Hilbert	98.9 %	99.8 %	99.7 %	79.0 %	80.2 %	81.7 %
		RankNet	98.9 %	99.8 %	99.8 %	82.0 %	81.7 %	82.3 %
		Maximum Diff	-	-	-	+3.0 %P	+2.2 %P	+2.1 %P
	UNSW-NB15	Base	76.1 %	79.7 %	79.8 %	80.8 %	80.9 %	81.1 %
		Hilbert	75.9 %	79.9 %	79.7 %	79.3 %	80.7 %	80.3 %
		RankNet	78.4 %	79.2 %	80.7 %	77.7 %	80.6 %	82.4 %
		Maximum Diff	+2.3 %P	-	-	-	-	+1.3 %P
	CIC-IDS2017 (multiclass classification)	Base	95.4 %	98.0 %	98.6 %	95.7 %	98.4 %	98.6 %
		Hilbert	95.0 %	98.5 %	99.0 %	95.9 %	98.5 %	98.7 %
		RankNet	95.5 %	98.6 %	98.8 %	96.5 %	98.6 %	98.8 %
		Maximum Diff	-	-	-	-	-	-
F1-score	NSL-KDD	Base	99.1 %	99.7 %	99.7 %	78.2 %	78.6 %	79.5 %
		Hilbert	98.8 %	99.7 %	99.7 %	78.0 %	79.5 %	81.6 %
		RankNet	98.9 %	99.7 %	99.8 %	81.9 %	81.4 %	82.0 %
		Maximum Diff	-	-	-	+3.7 %P	+2.8 %P	+2.5 %P
	UNSW-NB15	Base	77.4 %	78.8 %	79.0 %	84.7 %	84.9 %	85.1 %
		Hilbert	77.6 %	79.4 %	78.1 %	83.2 %	84.5 %	83.8 %
		RankNet	77.8 %	80.3 %	79.7 %	81.0 %	84.4 %	86.5 %
		Maximum Diff	-	+1.5 %P	-	-	-	+1.4 %P
	CIC-IDS2017 (multiclass classification)	Base	94.3 %	97.7 %	98.4 %	95.6 %	98.0 %	97.8 %
		Hilbert	93.9 %	98.3 %	98.9 %	95.8 %	98.3 %	98.6 %
		RankNet	94.6 %	98.5 %	98.7 %	96.1 %	98.3 %	98.5 %
		Maximum Diff	-	-	-	-	-	-



(a) CNN1 - Accuracy/epoch (b) CNN2 - Accuracy/epoch (c) CNN3 - Accuracy/epoch

(d) CNN1 - Loss/epoch (e) CNN2 - Loss/epoch (f) CNN3 - Loss/epoch

Fig. 3. UNSW-NB15 training result graph of training data. (x-aixs: epochs)
(a)~(c) accuracy graph (d)~(f) loss graph (legend: Basic, Hilbert, RankNet)

본방법 대비 속성 재배열 방법이 오히려 정확도가 ±0.5%P 차이만 보였다. 하지만, 훈련이 끝난 모델을 평가할때는 기본방법 대비 랭크넷의 속성 재배열 방법이 최대 3.7%P에서 최소 2.5%P의 F1스코어가 높았다.

4. 결론 및 향후연구

본 연구는 기존의 CNN IDS 연구들에서 CNN IDS 모델을 훈련할 때 훈련데이터의 속성 순서를 변경없이 그대로 배치하여 학습하는 것이 분류 정확도 성능 측면에서 최적이 아님을 실험을 통해 보이는 것을 주된 목적으로 한다. 이를 위해, CNN IDS 모델의 전처리 과정에서 데이터의 속성을 재배열하는 세 가지 방법(기본속성 배열, 랭크넷, 힐베르트 곡선),을 활용하여 세 개의 CNN 모델(CNN1, CNN2, CNN3)을 훈련하고 정확도와 F1스코어를 지표로 하여 성능을 비교했다. 실험한 결과를 보면, NSL-KDD 데이터세트로 훈련한 CNN1 모델에서 랭크넷 방법을 적용했을 때 정확도가 최대 3.7%P 높아짐을 확인했다. 이를 통해, 데이터의 속성의 순서를 재배열하면 CNN 모델의 성능을 향상시킬 수 있다는 것을 알 수 있으며 기본 순서 배열방법은 최적의 방법이 아님을 확인하였다.

향후 연구계획은 다음과 같다. 본 연구에서는 연구 목적에 따라 속성 재배열 방법을 적용한 데이터로 훈련하면 모델의 성능을 향상할 수 있다는 것을 확인하였으나, 어떤 속성이 IDS의 정확도 성능에 영향을 미치는지 또는 최적의 속성 재배열 방법을 찾지는 못했다. 따라서, 향후 연구에서는 추가적인 분석과 실험을 통해 최적의 성능을 갖는 속성 배열 방법에 대한 연구를 수행할 계획이다.

References

[1] Khan, A., Sohail, A., Zahoora, U., & Qureshi, A. S., "A survey of the recent architectures of deep convolutional neural networks." *Artificial intelligence review*, 53.8, pp.5455-5516, 2020.
DOI: https://doi.org/10.1007/s10462-020-09825-6

[2] Vinayakumar, R., Soman, K. P., & Poornachandran, P., "Applying convolutional neural network for network intrusion detection." *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. IEEE, pp.1222-1228, Sep. 2017.
DOI: https://doi.org/10.1109/ICACCI.2017.8126009

[3] Zhang, X., Ran, J., & Mi, J., "An intrusion detection system based on convolutional neural network for imbalanced network traffic." *2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT)*. IEEE, pp.456-460, Oct. 2019.
DOI: https://doi.org/10.1109/ICCSNT47585.2019.8962490

[4] Blanco, R., Cilla, J. J., Malagón, P., Penas, I., & Moya, J. M., "Tuning cnn input layout for ids with genetic algorithms." *International Conference on Hybrid Artificial Intelligence Systems. Springer, Cham*, pp. 197-209, June 2018.
DOI: https://doi.org/10.1007/978-3-319-92639-1_17

[5] Ren, Z., Chen, G., & Lu, W., "Space filling curve mapping for malware detection and classification." *Proceedings of the 2020 3rd International Conference on Computer Science and Software Engineering*. pp.176-180, May 2020.
DOI: https://doi.org/10.1145/3403746.3403924

[6] Luo, Y., Qin, X., Chai, C., Tang, N., Li, G., & Li, W., "Steerable self-driving data visualization." *IEEE Transactions on Knowledge and Data Engineering*, 34.1, pp.475-490, 2020.
DOI: https://doi.org/10.1109/TKDE.2020.2981464

[7] Lindsay, G. W., "Convolutional neural networks as a model of the visual system: Past, present, and future." *Journal of cognitive neuroscience*, 33.10, pp.2017-2031, 2021.
DOI: https://doi.org/10.1162/jocn_a_01544

[8] Bay, S. D., Kibler, D., Pazzani, M. J., & Smyth, P., "The UCI KDD archive of large data sets for data mining research and experimentation." *ACM SIGKDD explorations newsletter*, 2.2, pp.81-85, 2000.
DOI: https://doi.org/10.1145/380995.381030

[9] Moustafa, N., & Slay, J., "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)." *2015 military communications and information systems conference (MilCIS)*. IEEE, pp.1-6, Nov. 2015.
DOI: https://doi.org/10.1109/MilCIS.2015.7348942

[10] Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A., "Toward generating a new intrusion detection dataset and intrusion traffic characterization." *4th International Conference on Information Systems Security and Privacy (ICISSP)*, 1, pp.108-116, Jan. 2018.
DOI: http://doi.org/10.5220/0006639801080116

[11] Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., & Hullender, G., "Learning to rank using gradient descent." *Proceedings of the 22nd international conference on Machine learning*. pp.89-96, Aug. 2005.
DOI: https://doi.org/10.1145/1102351.1102363

[12] Khan, A., Sohail, A., Zahoora, U., & Qureshi, A. S., "A survey of the recent architectures of deep convolutional neural networks." *Artificial intelligence review*, 53.8, pp.5455-5516, 2020.
DOI: https://doi.org/10.1007/s10462-020-09825-6

박 휘 랑(Hweerang Park) [준회원]

- 2010년 : 전남대학교 졸업 (학사)
- 현재 : 국방대학교 국방관리대학원 국방과학학과 컴퓨터공학/사이버전협동전공 석사과정

〈관심분야〉
인공지능 보안, 적대적 머신러닝

조 영 호(Youngho Cho) [정회원]

- 1998년 2월 : 공군 사관학교 졸업 (학사)
- 2006년 2월 : 연세대학교 졸업 (공학석사)
- 2013년 2월 : University of Maryland, College Park USA 졸업 (공학박사)
- 현재 : 국방대학교 국방관리대학원 국방과학학과 컴퓨터공학/사이버전협동전공 부교수

〈관심분야〉
네트워크 보안, 스테가노그래피 봇넷, 신뢰 메커니즘, 블록체인, 디지털 포렌식, AI 보안 등