네트워크 침입탐지를 위한 딥러닝 모델 구조 최적화

금락운, 이청준, 권혁민, 최희열
한동대학교 전산전자공학부
e-mail : {21400047, 21400608, 21100047, hchoi}@handong.edu

Optimizing Deep Learning Model Architecture for Network Intrusion Detection

Rakun Keum, Chungjun Lee, Hyuckmin Kwon, Heeyoul Choi
School of Computer Science and Electrical Engineering, Handong Global University

요 약

네트워크 침입탐지 시스템(Network Intrusion Detection System)은 데이터를 기반으로 침입을 탐지하여 사이버 보안에서 중요한 역할을 수행한다. 최근 딥러닝 기술이 주목 받기 시작하면서 Deep Neural Network(DNN) 알고리즘을 이용하여 NIDS 를 구현한 연구사례들이 있다. 그러나 해당 논문들은 딥러닝 모델의 구현에 초점을 두었기 때문에 최적의 성능을 도출하지 못했다. 특히 최적화 알고리즘(Optimizer)과 학습률(Learning Rate)조정을 통해서 모델을 더 개선할 수 있는 가능성이 남아있다. 본 논문에서 구현한 모델은 NSL-KDD 데이터셋 Binary Classification 에서 82.5%의 Accuracy 를 달성하여 기존 논문들의 결과보다 성능을 제고하였다.

1. 서론

인터넷 네트워크의 보급이 증가하고 이를 통한 정보유통이 증가함에 따라 네트워크 해킹으로 인한 피해도 증가하고 있다 [1]. 네트워크 침입탐지 시스템 (Network Intrusion Detection System)은 방화벽과 더불어 네트워크 보안에 있어 중요한 역할을 담당하고 있다. NIDS 는 포트번호, 프로토콜, 트래픽 크기 등 네트워크 상의 데이터를 기반으로 침입을 탐지한다 과거에 NIDS 는 전문가의 지식에 기반하여 만들어 지는 것이 보편적이었지만, 근래에는 방대한 양의 데이터를 기반으로 머신러닝을 적용하는 경우가 늘고 있다. 특히 최근 딥러닝 기술이 주목 받기 시작하면서 NIDS 에 Deep Neural Network(DNN) 알고리즘을 적용한 연구사례들이 있다[2]. 그러나 해당 논문들은 알고리즘의 적용에 초점을 두었기 때문에 구현은 성공하였지만 최적화를 통한 성능 개선의 여지를 남겨두고 있다. 본 연구는 실험을 통해 최적의 Parameter 를 찾아내어 기존 논문들보다 성능을 제고하는 것을 목표로 하였다.

2. 관련연구

관련연구는 NSL-KDD 데이터셋을 기반으로 NIDS 에 DNN 알고리즘을 적용한 기존 논문들을 조사하였다. 먼저 2017 년에 딥러닝 알고리즘을 이용하여 Deep Neural Network(DNN) 모델과 Recurrent Neural Network(RNN) 모델을 구현한 사례가 있다[3]. 이 논문에서는 DNN, RNN 모델이 각각 73.1%와 81.3%의 Accuracy 를 보였다. 또 다른 연구사례에서는 현재 새롭게 대두되고 있는 네트워크 아키텍처인 Software Defined Network(SDN) 환경을 고려하여 DNN 을 적용하였고 75.8%의 Accuracy 를 보였다[4].

3. 데이터셋(dataset)

3.1 Dataset 특징

우리는 NIDS 를 구현하기 위해 NSL-KDD 데이터셋을 사용하여 DNN 모델을 학습하였다. NSL-KDD 데이터셋은 표 1 에 기술된 것처럼 크게는 DoS, Probe, R2L, U2R 로 총 4 가지의 공격유형과 정상 레코드로 이루어져 있고 각 공격 유형에 해당하는 데이터는 세부 공격유형으로 구분될 수 있다. NSL-KDD 데이터셋의 Feature 들은 Nominal Feature 3 개, Binary Feature 6 개, Numeric Feature 32 개로 총 41 개의 Feature 로 이루어져 있다. Train 데이터셋과 Test 데이터셋은 각각 125,973, 22,544 개의 레코드로 이루어져 있다 [5].

표 1. NSL-KDD 데이터셋 데이터 분포

Dataset	Train	Test
Total Records	125,973	22,544
Normal	67,343(53.46%)	9,711(43.08%)
DoS	45,927(36.46%)	7,458(33.08%)
Probe	11,656(9.25%)	2,421(10.74%)
U2R	52(0.04%)	200(0.89%)
R2L	995(0.79%)	2,754(12.22%)

3.2 데이터 전처리

먼저 NSL-KDD dataset 에서 'TCP' 또는 'UDP' 과 같은 Nominal 형태의 Feature 들에 대해서 Label Encoding 을 적용하여 각각의 값들을 자연수로 치환하였다. 다른 Feature 에 대해서는 최대값과 최소값

을 구하여 정규화 하는 Min-Max Normalization 을 적용하였다.

4. 실험

4.1 모델 구조

본 논문에서 제안하는 Deep Neural Network(DNN) 모델의 구조는 아래의 표 2 와 같이 Input Layer, Hidden Layer, Output Layer 로 구성된다. Input Layer 는 NSL-KDD 의 Feature 개수와 대응하는 41 이다. Hidden Layer 는 3 개 층을 쌓았으며 각 층은 300, 100, 50 의 Hidden 노드의 갯수를 가진다. Output Layer 는 Binary Classification, Multi Classification 에서 각각 2 와 5 값을 가진다. 또한 활성화 함수(Activation Function)로 Leaky ReLU 를 사용하였다.

표 2. Deep Neural Network 모델의 구조

Algorithm	Input Layer	Hidden Layer	Output Layer	Activation
DNN	41	300, 100, 50	2(binary)/5(multi)	Leaky-ReLU

4.2 실험 과정

먼저 Train 데이터셋을 Sub-Train Set 과 Validation Set 으로 각각 90%, 10% 비율로 무작위 추출하였다. 다음으로는 DNN 모델을 만들어서 각 Epoch 마다 Sub-Train Set 을 통해 모델을 학습시키고, Validation Set 을 통해 모델의 학습 정도를 평가하였다. 학습할 때 Max Epoch 은 10,000 으로 설정하였으며 Early-Stopping 기법을 이용하여 특정 Patience 에 도달하면 더 이상 학습이 되지 않는 것으로 간주하여 학습을 종료시켰다. 학습과정을 마친 후 Test 데이터셋을 사용하여 모델의 성능을 평가하였다. 최종 결과 값은 실험 과정을 15 번 반복하여 평균값을 도출하였다.

최적화를 위해 실험한 Parameter 들은 최적화 알고리즘(Optimizer)과 학습률(Learning Rate)이다. 딥러닝에서 대표적으로 사용되는 RMSprop, Adam, SGD Optimizer 에 대하여 실험을 진행하였다. 위의 세 가지 최적화 알고리즘과 학습률에 대한 Accuracy 를 도출하였다.

4.3 실험 결과

NSL-KDD 데이터셋에 대한 DNN 모델 실험 결과를 표 3 과 표 4 에 정리 하였다. 최적의 모델은 Binary Classification 에서 82.2%의 Accuracy 를 보였고, Multi-Class Classification 에서는 77.3%의 Accuracy 를 달성하였다.

표 3. Multi-Class Accuracy

Learning rate	RMSprop	Adam	SGD
0.1	*0.740*	*0.743*	*0.760*
0.01	*0.751*	*0.749*	***0.773***
0.001	*0.760*	*0.763*	*0.769*
0.0001	*0.753*	*0.768*	*0.631*

표 4. Binary Class Accuracy

Learning rate	RMSprop	Adam	SGD
0.1	*0.780*	*0.775*	***0.822***
0.01	*0.782*	*0.795*	*0.817*
0.001	*0.796*	*0.795*	*0.771*
0.0001	*0.812*	*0.779*	*0.742*

5. 결론

본 논문에서는 NSL-KDD Dataset 을 기반으로 DNN 알고리즘을 통해 NIDS 의 성능을 실험하였다. 실험을 통해서 모델 Parameter 의 최적화를 통해 기존 연구 사례보다 모델의 성능을 개선할 수 있었다. 향후 과제로는 Software Defined Netwrok(SDN)이나 사물인터넷 네트워크 등 특정 환경에 맞추어 구현된 DNN 모델을 최적화 하여 성능을 개선할 수 있을 것으로 기대된다.

6. 사사

이 논문은 2018 년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원(No.2018-0-00749,인공지능 기반 가상 네트워크 관리기술 개발)과, 과학기술정보통신부와 정보통신기술진흥센터의 소프트웨어중심대학 지원사업(2019-0-00130)의 지원을 받아 수행하였음.

참고문헌

[1] R. Doshi, N. Apthorpe, and N. Feamster, “Machine Learning DDoS Detection for Consumer Internet of Things Devices,” *IEEE Security and Privacy Workshops (SPW) 2018.*

[2] P. Garcia-Teodoro, J. Diaz-Verdego, G.Macia-Fernandez, and E. Vazquez, “Anomaly-based Network Intrusion Detection: Techniques, Systems and Challenges,” *Computers and Security Volume 28 Issue 1-2, February 2009, Pages 18-28.*

[3] C. Yin, Y. Zhu, and X. He, “A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Network,” *IEEE Access, Volume 5. Publication October 12, 2017.*

[4] T. Tang, L. Mhamdi, D. McLernon, S.A.R. Zaidi, and M. Ghogho, “Deep Learning Approach for Network Intrusion Detection in Software Defined Networking,” *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM).*

[5] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, “A Detailed Analysis of the KDD CUP 99 Data Set,” *Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA 2009).*