텍스트 오토인코더 기반 네트워크 침입 탐지에 적합한 토큰화 및 임베딩 조합에 관한 연구

박주혜, 최일호, 오현식, 전슬기
롯데정보통신 AI 기술팀

joohyepark@lotte.net, ilho.choi@lotte.net, hyunsikoh@lotte.net, seulki.jun@lotte.net

A Study on the combination of tokenization and embedding suitable for text autoencoder based network intrusion detection

Park Joo Hye, Choi Il Ho, Oh Hyun Sik, Jun Seul Ki
Lotte Data Communication Co., Ltd

요 약

4 차 산업혁명이 도래하고 인공지능과 빅데이터 기술이 급속도로 발전하면서 그 중요성이 커지고 있다. 그로 인해 데이터의 수집과 함께 데이터를 외부의 공격으로부터 보호하는 보안 기술에 대한 중요성이 부각되고 있다. 규칙 기반의 보안 방법은 규칙 이외의 공격 발생시 탐지하기 어렵고 지도학습 모델의 경우 실제 보안 데이터의 불균형 문제로 일반화된 모델의 구축이 어렵다. 이러한 방법들의 단점을 극복하고자 비지도 학습인 텍스트 분석 기법 기반 오토인코더를 활용하고 있으나 로그 데이터의 형태가 일반적인 텍스트 데이터와 달라 토큰화와 임베딩 과정의 어려움이 있다. 본 연구에서는 보안 로그 데이터에 텍스트 분석 기법 기반 오토인코더를 적용하고 일반적인 토큰화 방법과 임베딩 방법의 조합을 실험하여 효율성이 좋은 조합을 도출하고자 하였다. 실험 결과 토큰화 방법과 상관없이 단순 단어 수준 임베딩 방법이 성능이 높게 나오는 것을 확인하였다.

Ⅰ. 서 론

4 차 산업혁명에 접어들면서 인공지능, 빅데이터 기술이 발전함과 동시에 그 중요성이 커지고 있다[1]. 다양한 산업 분야에서 인공지능을 활용하여 효율성과 생산성 개선을 시도하고 있다. 그로 인해 기업은 인공지능의 학습을 위해서 개인정보가 포함된 대용량의 데이터를 수집하고 축적하는 것뿐만 아니라 외부의 공격으로부터 보호해야할 의무를 가지게 되었다.

기업에서 사용하는 기존 보안 장비로는 IDS, WAF 등이 있다. IDS 장비 탐지 방법으로는 대표적으로 서명 기반 탐지, 이상 징후 기반 탐지, 상태 유지 프로토콜 분석 3 가지가 있다. 서명 기반 탐지는 네트워크를 통과하는 패킷을 모니터링 하여 “서명”을 비교하여 공격여부를 판단한다. 이상 징후 기반 탐지는 포트, 대역폭, 프로토콜 등의 장치에 대하여 정상 기준을 정하고 네트워크 트래픽이 기준을 넘는 경우 공격으로 규정한다. 상태 유지 프로토콜은 정상 프로토콜을 이용해 정상 프로필을 사전에 정의하고 각 프로토콜에서 발생하는 이벤트를 사전에 정의된 것과 비교하여 공격 여부를 판단한다. WAF 장비는 화이트, 블랙리스트를 정의하여 공격을 방어하는 방법을 사용한다. 이처럼 기존 장비의 경우 규칙기반의 알고리즘을 이용하여 위협을 탐지한다. 그러나 규칙 기반 탐지는 규칙 이외의 공격유형에 대처가 어렵다. 딥러닝 기술 중 지도학습은 데이터를 구하기 어렵고 데이터를 확보하여도 불균형이 심해 학습을 하는 것에 어려움이 있다. 이러한 문제점들을 해결하고자 비지도 학습 중 텍스트 오토인코더를 활용해 분석을 시도하고 있으나 보안 로그 데이터가 일반적인 텍스트 데이터와 형태가 달라 토큰화와 임베딩에 어려움이 있다. 이에 본 연구에서는 텍스트 오토인코더 기법에 일반적인 토큰화, 임베딩 기법의 조합을 실험하고 가장 효율적인 조합을 찾고자 한다.

Ⅱ. 적층 텍스트 오토인코더

본 연구는 텍스트로 구성되어 있는 보안 로그 데이터의 토큰화 방법과 임베딩 방법이 모델의 성능에 영향을 어떻게 미치는가에 초점을 맞추고 있다. 따라서 실험을 위하여 적층 오토인코더 모델로 고정하여 진행한다. 적층 오토인코더를 선택한 것은 일반 오토인코더 모델에 비해서 레이어수가 많아 텍스트 데이터의 패턴의 특징을 더 잘 추출할 수 있기 때문이다[2]. 적층 오토인코더 구조를 간단히 표현한 그림은 그림 1 과 같다.

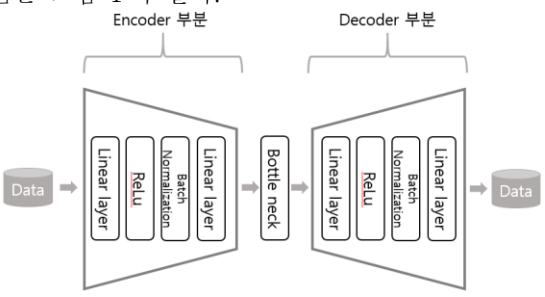

그림 1

본 연구에서 사용한 적층 오토인코더는 임베딩된 로그데이터를 입력으로 받아서 6 개의 layer 를 통과하며 차원을 축소하였다가 디코더에서 입력데이터의 개수로 복원한다. 모델의 기울기가 소실되어 학습이 안되는 것을 방지하기 위해 일반적으로 사용되는 활성함수인 ReLu 를 사용한다. 또한 각 레이어를 통과할 때 마다 분포의 차이가 발생하여 기울기가 소실되는 것을 방지하기 위해서 배치 정규화를 사용한다. 실험에 사용한 파라미터 값은 표 1 과 같다.

표 1

하이퍼파라미터 명	하이퍼파라미터 값
학습률	0.001
최적화함수	Adam
배치사이즈	256
Epoch	10

Ⅲ. CSIC2010 데이터셋

본 연구에서는 CSIC 의 정보 보안 연구소에서 개발된 CSIC2010 보안 데이터 세트를 사용하여 모델 학습과 테스트를 수행한다. CSIC2010 데이터 원본은 Normal Train 36,000 개, Normal Test 36,000 개, Anomalous Test 25,065 개로 구성되어 있다. 오토인코더 모델에서 정상과 공격의 기준이 되는 임계값을 설정하기 위해서 검증 데이터 셋이 필요하므로 Normal Test 데이터에서 10,800 개의 데이터를 분리하여 검증을 위한 데이터 셋을 구성하고 나머지 25,200 개의 데이터를 테스트 셋으로 구성하였다. 본 연구에서는 로그데이터가 가지고 있는 텍스트 형태의 비정형 데이터에 대한 분석을 하는 것이므로 CSIC2010 데이터가 가지고 있는 Feature 들 중 'expert message', 'content' 부분만 추출하고 결합하여 사용한다.

Ⅳ. 실험

본 연구에서 사용하는 토큰화 실험은 5 가지이고 임베딩 방법은 총 3 가지이다. 토큰화 방법은 '특수기호를 기준으로 단순 토큰화', 'BPE[3] 알고리즘을 이용한 토큰화', '캐릭터 단위의 토큰화'[4], 총 3 가지 방법에 BPE, 캐릭터 단위 토큰화 방법에서 특수기호를 제거한 경우와 제거하지 않은 경우로 나누어 5 가지 방법을 실험하였다. 임베딩 방법으로는 토큰화된 단어에 정수 인덱스를 부여하고 임베딩 레이어를 이용하는 '단순 토큰 단위 임베딩', 'bag of words' 방법, 'Word2vec' 방법을 이용했다. 토큰화 방법과 임베딩 방법을 조합하여 총 15 가지의 실험을 진행하였다.

Ⅴ. 성능 평가

표 2

조합	정확도	정밀도	재현율	F1
특수기호+ Simple	0.322	0.322	1	0.499
특수기호+ BOW	0.616	0.457	0.825	0.588
특수기호+ W2V	0.78	0.627	0.829	0.714
BPE(포함)+ Simple	0.821	0.661	0.946	0.778
BPE(포함)+ BOW	0.804	0.629	0.998	0.771
BPE(포함)+ W2V	0.649	0.486	0.967	0.647
BPE(제거)+ Simple	0.85	0.7	0.958	0.809
BPE(제거)+ BOW	0.793	0.617	0.993	0.761
BPE(제거)+ W2V	0.568	0.434	0.982	0.601
캐릭터(포함)+ Simple	0.835	0.674	0.975	0.797
캐릭터(포함)+ BOW	0.794	0.618	0.998	0.763
캐릭터(포함)+ W2V	0.643	0.481	0.936	0.635
캐릭터(제거)+ Simple	0.817	0.648	0.987	0.782
캐릭터(제거)+ BOW	0.774	0.596	0.993	0.745
캐릭터(제거)+ W2V	0.658	0.491	0.813	0.612

본 연구는 5 가지 토큰화 방법을 3 가지 임베딩 방법을 이용하여 모델을 테스트하고 각 성능을 비교하여 적합한 토큰화 방법과 임베딩 방법의 조합을 찾는 것을 목적으로 한다. 각 조합의 성능 평가는 정확도, 재현율, 정밀도, F1-Score 를 고려하여 종합적으로 평가한다. 실험의 결과는 도표 2 와 같다. 특수 기호 기준으로 토큰화 한 경우를 제외하고는 대부분 단순 단어 임베딩을 적용한 것이 대체로 성능이 잘 나오는 것을 확인할 수 있다.

Ⅵ. 결론

본 연구에서는 텍스트 분석 기법을 활용한 오토인코더 모델에서 이용할 수 있는 토큰화 방법과 임베딩 방법을 조합하여 실험해 보고 그 효율성을 검증하였다. 기존 텍스트 분석과 같이 단어의 의미 단위로 토큰화 하는 것은 배제하고 크게 3 가지 토큰화 방법을 적용하였으며 세부적으로 특수 기호의 제거 여부를 구분하여 총 5 가지 방법을 적용하였다. 임베딩 기법은 텍스트 분석에서 기본적으로 사용되는 모델 중 3 가지를 선택하여 적용하였다. 실험 결과 토큰화 방법과 상관없이 단순 단어 임베딩의 성능이 높게 나온 것을 확인하였다. 실험 결과로 볼 때 토큰화의 방법은 성능의 큰 영향을 미치지 않는 것으로 볼 수 있는데 이것은 토큰화 방법이 모델에 영향을 줄만큼 정교하지 못했기 때문이라고 볼 수 있다. 향후 연구에서는 로그 데이터 분석만을 위한 토큰화 기법을 구현하고 성능 향상 여부를 확인해 볼 필요성이 있다.

ACKNOWLEDGMENT

본 논문은 롯데정보통신 AI 기술팀과 보안 플랫폼팀의 내부 연구 프로젝트로 진행된 연구이다.

참 고 문 헌

[1] 정동규, 송도선, "인공지능과 사물인터넷 특징 및 결합 산업 동향", 한국정보기술학회지 제 15 권 제 2 호 29-39.

[2] Yu yan., et al., "A Network Intrusion Detection Method Based on Stacked Autoencoder and LSTM", ICC, 6, 2020

[3] Rico Senrich., et al., "Neural Machine Translation of Rare Words with Subword Units", Proceedings of the 54th Annual Meeting of the Association for Computational linguistics, Volume 1, 2016

[4] Monika Arora, et al., "Character level embedding with deep convolutional neural network for text normalization of unstructured of unstructured data for Twitter sentiment analysis", Social Network Analysis and Mining, 3, 2019