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

Word2Vec과 가속화 계층적 밀집도 기반 클러스터링을 활용한 효율적 봇넷 탐지 기법☆

An Efficient BotNet Detection Scheme Exploiting Word2Vec and Accelerated Hierarchical Density-based Clustering

이 태 일[1] 김 관 현[1] 이 지 현[1] 이 수 철[1*]
Taeil Lee Kwanhyun Kim Jihyun Lee Suchul Lee

요 약

수많은 기업체, 기관, 개인 사용자가 대규모 DDos(Distributed Denial of Service)공격에 의한 피해에 노출되고 있다. DDoS 공격은 좀비PC라 불리는 수많은 컴퓨터들과 계층적 지령구조를 좀비PC들을 제어하는 네트워크인 봇넷을 통하여 수행된다. 통상의 악성코드 탐지 소프트웨어나 백신은 멀웨어를 탐지하기 위해서 사전에 심층 분석을 통한 멀웨어 시그니처를 밝혀야 하며, 이를 탐지 소프트웨어나 백신에 업데이트하여야 한다. 이 과정은 방대한 시간과 비용이 소모된다. 본고에서는 인공신경망 모델을 이용하여 주기적인 시그니처 사전 업데이트가 필요 없는 봇넷 탐지기법을 제안한다. 제안하는 인공신경망 모델은 Word2Vec과 가속화 계층적 밀집도 기반 클러스터링을 활용한다. 제안기법의 봇넷 탐지성능은 CTU-13 데이터셋을 이용하여 평가하였다. 성능평가 결과, 분류 정확도 99.9%로 기존 방법에 비해 우수한 멀웨어 탐지율을 보인다.

☞ 주제어 : 봇넷 탐지, Word2Vec, 클러스터링, Skip-gram기법

ABSTRACT

Numerous enterprises, organizations and individual users are exposed to large DDoS (Distributed Denial of Service) attacks. DDoS attacks are performed through a BotNet, which is composed of a number of computers infected with a malware, e.g., zombie PCs and a special computer that controls the zombie PCs within a hierarchical chain of a command system. In order to detect a malware, a malware detection software or a vaccine program must identify the malware signature through an in-depth analysis, and these signatures need to be updated in priori. This is time consuming and costly. In this paper, we propose a botnet detection scheme that does not require a periodic signature update using an artificial neural network model. The proposed scheme exploits Word2Vec and accelerated hierarchical density-based clustering. Botnet detection performance of the proposed method was evaluated using the CTU-13 dataset. The experimental result shows that the detection rate is 99.9%, which outperforms the conventional method.

☞ keyword : BotNet Detection, Word2Vec, Clustering, Skip-gram

1. 서 론

인터넷의 발달은 인류에게 편리함을 가져다주었지만 동시에 악성코드와 같은 또 다른 문제를 야기하였다. 이로 인해 현재 사이버 공간에서는 수많은 위협이 산재해 있다. 예컨대, 제3자의 개인 정보를 갈취하여 금전적인 이익을 보거나, 경쟁사의 IT 서비스를 방해하는 등 인터넷상의 위협요인들은 다양한 형태로 존재하며, 새로운 공격기법들이 개발되고 있으며, 이를 모두 탐지하기는 거의 불가능하다고 하겠다.

DDoS공격은 지난 20여 년간 항상 수행되어 왔다. DDoS 공격은 좀비PC라 불리는 수많은 컴퓨터들과 계층적 지령구조에 의해 이들을 제어하는 C&C(Command and Control)서버로 구성된 네트워크인 봇넷을 통하여 수행된다. DDoS공격은 현존하는 악성코드 탐지소프트웨어나 백신을 이용하여 탐지하기가 매우 어렵다. 왜냐하면 통상의 악성코드 탐지 기법은 시그니처에 기반을 두는데, DDoS공격을 수행하는 악성코드 내에서 특정 시그니처를

1 Dept. of Computer Science and Information Engineering, Korea National University of Transportation, Uiwang, Kyunggi, 16106, Korea.
* Corresponding author (sclee@ut.ac.kr)
[Received 20 May 2019, Reviewed 29 May 2019(R2 23 August 2019), Accepted 25 October 2019]
☆ 이 성과는 2019년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No.2017R1C1B5017028). 이 연구는 2019년 한국교통대학교 지원을 받아 수행하였음.

포함하지 않는 경우도 다반사이며 특정 시그니처가 존재한다손 치더라도 공격자는 이를 계속 변화, 진화시킬 것이다. 통상적으로 특정 시그니처를 밝혀내는 데에는 악성코드 채증, 채증된 악성코드의 심층 분석, 분석을 통해 최종적으로 특정된 시그니처를 악성코드 탐지 소프트웨어나 백신에 업데이트하여야 하므로 방대한 시간과 비용이 소모된다. 따라서 신속하게 DDoS공격을 무력화 시키는 방법은 실질적으로 존재할 수 없다고 하겠다. 심지어 시그니처 기반 방법으로 제로데이 DDoS공격에 대한 신속한 탐지는 거의 불가능한 문제가 된다.

본고에서는 DDoS 공격 시도, 패킷, 페이로드 가로채기 및 대상 노드 크래킹과 같은 악성 공격에 대해 효과적으로 대응하기 위하여 통상적인 멀웨어 바이너리에서 시그니처를 분석, 추출하는 방식과 달리 트래픽의 송수신지 등과 네트워크 정보를 기반으로 시그니처를 생성하는 방법을 제안한다. 시그니처 생성에 [1] 있어 5-tuple 중 송신지 포트를 제외한 4-tuple(송신지 IP 주소, 수신지 IP 주소, 프로토콜, 수신지 포트번호)로 워드 임베딩(word embedding)을 구성하고 이를 자연어 학습 모델인 Word2Vec의 [2] [3] Skip-gram 알고리즘을 [4] 활용한 봇넷 탐지 방법을 제안한다.

[5]에서는 기계어 명령어의 Word2Vec 워드 임베딩을 활용한 실험 결과를 분석하여 최적의 하이퍼 파라미터값을 찾으면 높은 F-점수(F-Score)를 얻을 수 있음을 확인하였다. [4]에서는 Word2Vec 인공신경망 Skip-gram 알고리즘을 멀웨어 시그니처 추출기법으로 활용하였다. 본 연구에서는 기존 연구와 달리 봇넷 탐지를 위해 네트워크 정보를 기반으로 워드 임베딩을 시도하였다. CTU-13 데이터셋을 [6] 활용한 실험결과, 0.95의 F-점수, 0.91의 정밀도를 보였다. 제안기법은 비지도 학습(unsupervised learning)에 기반을 두므로 데이터의 레이블의 존재 여부에 관계없이 적용할 수 있다는 장점이 있다. 따라서 공격에 대한 사전 지식의 필요하지 않아 범용성 측면에서 우수하다고 하겠다.

본 논문의 공헌은 다음과 같다. 첫째, 봇넷 탐지에 Word2Vec의 Skip-gram 알고리즘을 적용하였으며, 제안 기법의 탐지 성능을 실제의 봇넷 데이터인 CTU-13 데이터셋을 통해 실험적으로 검증하였다. 둘째, 제안기법에서는 봇넷의 동작 특성을 고려한 워드 임베딩 방법을 적용하였다. 마지막으로 제안기법의 단어 벡터 간 상관관계를 확인하고 분류 성능을 직관적으로 확인할 수 있도록 t-SNE(t-Stochastic Neighbor Embedding) [7] 및 DBScan(Density-based spatial clustering of applications with noise)기법을 [8] 통해 단어 벡터를 시각화하여 봇넷 클러스터의 효과적인 형성을 입증하였다.

본고의 구성은 다음과 같다. 2장에서는 관련 연구를 요약한다. 3장에서는 CTU-13 데이터셋을 활용한 연구에 대한 요약과 봇넷 트래픽, 정상 트래픽, 백그라운드 트래픽들이 혼용되어 있는 특징의 정의와 활용방안에 관해서 기술한다. 4장에서는 제안기법의 학습 과정을 상세히 기술한다. 5장에서는 실험환경 및 결과를 기술한다. 마지막으로 6장에서 본고를 맺는다.

2. 관련 연구

인공지능 등 기계학습 기반 멀웨어 분석 및 분류기술은 전처리 단계를 거치게 된다. 통상적으로 기계학습 연구 분야에서는 이 과정을 특징 공학(feature engineering)이라 한다 [9]. 어떠한 특징(feature)을 어떠한 방법으로 가공 및 적용할 것인지에 관련된 연구 분야라고 볼 수 있다. 본 장에서는 네트워크 침입 탐지 시스템과 관련된 선행 연구는 [10]에 요약되어 있다. 본고에서는 행동기반 침입탐지연구와 관련하여 클러스터링(clustering), 트래픽 분류(traffic classfication), 자연어처리 (NLP : Natural Language Processing) 관련 연구에 대하여 살펴본다.

2.1 클러스터링 및 트래픽 분류

신경망 기반 클러스터링 알고리즘인 SOM (Self Organizing Maps)을 활용하여 다양한 시나리오에서 효과적으로 멀웨어를 탐지 할 수 있음이 입증되었다 [11][12][13]. KDD-99 데이터셋의 41개 특징에 대하여 테스트한 결과, 1.38%의 오탐율과 90.4%의 정탐율을 보였다 [14] . 또한 신경망 기반의 클러스터링 기법 중 기본적인 논리가 SOM과 매우 유사한 적응 공명 이론(ART : Adaptive Resonance Theory)은 27가지 공격 시나리오를 가진 네트워크 플로우 데이터셋의 IP, TCP, UDP 및 ICMP 헤더에서 추출한 27가지 특징을 추출하고 멀웨어 탐지성능을 평가하여 97%의 정확도를 보였다. CNN (Convolutional Neural Networks)은 세션 중에 정상적인 네트워크 트래픽의 바이트 시퀀스를 학습하고 이후에 네트워크 프로토콜별로 분류하는 것이 가능하다. H. Kim et al의 연구에서는 [15] 네트워크 플로우를 이미지화하여 CNN 모델의 훈련 데이터로 사용하는 방식으로 네트워크 침입을 탐지하는 방안을 제시하였다. 단방향 네트워크 플로우에서 발생하는 모든 특징들을 이미지화하기 때문에 특징 공학적인 측면에서 전처리 과정

의 시간 소요를 줄여주는 데에 의의가 있다.

2.2 자연어처리

Bellekens et al의 연구는 [16] 침입 탐지를 위한 신경망 기반의 NLP(Natural Language Processing) 알고리즘을 활용했다. 문서 수준의 임베딩(document-level embedding)을 생성하는 paragraph vector 알고리즘을 사용하여 계산에서의 용이성이 돋보인다. 제안된 방법은 C&C(Command & Control) 컴퓨터*의 탐지를 수행한다. 멀웨어의 통신을 탐지 대상으로 하여 악의적인 트래픽을 지도기계학습 방법의 하나인 SVM(Support Vector Machine)을 통해 분류한다. 이 방법은 학습 방법 및 테스트 데이터에 따라 다양한 정확도 0.77-0.99 및 리콜 0.69-1.00을 달성하였다. 하지만 이 방법의 단점은 기계학습과정에서 레이블 된 학습데이터를 사용해야 한다는 단점이 있으며, 어떠한 학습 데이터를 사용하느냐에 성능이 의존적이다. 반면 본 연구에서 제안하는 기법은 데이터에 구애받지 않고 비지도 학습을 기반으로 하므로 데이터의 레이블링을 필요로 하지 않는다. 따라서 침입 기법에 대한 사전 지식이 필요하지 않다.

R. Carrasco et al의 연구에서는 [4] 4개의 특징(송신지 IP주소, 수신지 IP주소, 프로토콜, 수신지 포트번호)에 대하여 Skip-gram기법을 활용하여 각 단어를 벡터화 하였다. 각 단어 벡터 사이의 코사인 유사도를 측정하여 비정상 트래픽을 탐지하였다. 본고에서는 [4] 와 달리 학습 과정에서 포트 번호가 동음이의어로 사용될 가능성이 많으며, 이로 인해 발생하는 탐지 성능 저하문제를 해결하기 위하여 전처리 과정을 변경하였다. 또한 비정상 트래픽을 더욱 명확하게 분류하기 위한 기법으로 밀도 기반 클러스터링(DBScan: Density Based Spatial Clustering of Application with noise)기법을 [8] 도입하였다.

3. 데이터와 전처리

3.1 CTU-13 데이터셋

본고에서 사용된 데이터셋은 2014년 S.Garcia가 공개한 CTU-13 데이터셋을 [6] 사용하였다. 해당 데이터셋은 총 13개의 서로 다른 봇넷 호스트를 포함하는 시나리오로 이루어져 있다. 게다가 봇넷 트래픽, 정상 트래픽, 백그라운드 트래픽들이 혼용되어있어 본 연구에 적합하다고 판단하였다. 또한 CTU-13 데이터셋은 시나리오별로 단방향 넷플로우파일과 침입탐지시스템(IDS : Intrusion Detection System)에서 채증한 pcap 형식의 파일을 제공한다. 그중에서도 봇 호스트의 개수가 가장 많은 10번 시나리오로 제공된 넷플로우 파일을 사용하여 연구를 진행하였다. 10번 시나리오의 내용은 전체 플로우 개수 1,309,791, 봇넷 플로우 개수 106,315(8.11%), 정상 플로우 개수 15,847(1.2%), C&C 플로우 개수 37(0.002%), 백그라운드 플로우 개수 1,187,592(90.67%)의 내용을 포함하고 있다.

10번 시나리오는 봇 호스트 10개가 한 개의 호스트에 대하여 UDP를 통하여 DDoS 공격을 진행하는 시나리오다. 총 13개의 시나리오 중 가장 높은 Bonnet Flows 106,315(8.11%)를 가지고 있으며, CTU-13 Dataset 10번 시나리오로 진행한 Songhui Ryu et al의 연구에서 [17] CTU-13 데이터셋의 다양한 시나리오로 멀웨어 탐지 성능 평가를 진행하였다. 예컨대, 4번, 10번,11번 시나리오에서 시나리오 4번이 가장 규모가 크지만 Rbot이 하나만 존재하였으며, 봇넷 트래픽 비율이 0.15 %이므로 데이터의 심각한 불균형을 나타내었다. 이는 오히려 제안기법이 도전적인 환경에서 봇넷을 탐지하는데 효과적인지를 평가하는 기준이 될 수 있으나, 성능평가 결과를 정량화 하는데 다소 어려움이 있어 제외하였다. 이에 반해, 시나리오 10번은 봇넷 트래픽의 8.11%, 시나리오 11번은 7.6%번으로 나타난 것을 확인하여, 본 연구에 봇넷 트래픽 비율이 높은 10번 시나리오를 적용하였다. 또한 10번 시나리오는 프로토콜의 다양성이 풍부하며 봇 호스트들이 다양한 DDoS 공격을 수행하는 형태를 보여 타 시나리오에 비해 실제 환경과 유사도가 더욱 높다.

3.2 전처리 : 특징선택

본고에서 수행하고자 하는 봇넷 탐지 및 일반적으로 인터넷 응용 트래픽 분류(Internet traffic classification)이라 [18][19] 불리는 연구에 있어서 5-tuple이 중요한 특징으로 작용한다는 점이 알려져 있으며, 이는 통상의 경우 네트워크의 성능 차이 등을 반영하는 특징들은 연결이 수립된 호스트간의 전송 선로에서의 성능 차이 값을 포함하고 있거나 값의 분포가 일관성이 없기 때문이다. 따라서 본고에서는 5-tuple을 위주로 특징을 선택하되 송신지 포트를 제외한 4-tuple - 송신지 IP 주소, 수신지 IP 주소, 프로토콜, 수신지 포트번호 - 을 기본 특징으로 선정하였다. 왜냐하면, CTU-13 데이터셋은 단방향 트래픽만을 기록하고 있어 송신지 IP 주소 간 유사도를 특징으로 선정한다. 다

* 계층적 지령구조를 가진 봇넷에서, 계층 최상위에 위치하여 좀비PC들에게 지령을 내리는 특수 목적의 컴퓨터

만 송신지 포트번호는 송신자가 직접 바인딩하지 않을 경우 랜덤하게 선택된 포트번호로 연결을 수립하므로 특징을 추출할 수 없고 대부분의 호스트는 소켓의 송신지 포트번호를 특정포트와 연관시키지 않아 불필요한 계산이 증가하는 문제점을 야기함을 실험적으로 확인하고 본고에서 제외하였다.

ICMP패킷 전처리 : 프로토콜 선정에서는 인터넷 통신의 기본이라고 할 수 있는 TCP, UDP, ICMP를 사용한다. 일반적으로 ICMP는 포트 번호 없이 근원지, 목적지 주소만을 사용하지만 포트 번호가 기재되어 있는 예외적인 경우가 있다. UDP 통신에서 목적지의 포트가 열려있지 않을 때, 목적지 호스트는 ICMP를 사용하여 포트 도달 불가(port unreachable)라는 에러 메시지를 전달한다. 에러 메시지 내부에는 통신에 사용되었던 한 쌍의 송/수신지 IP 주소와 포트 번호가 저장되어 있다. ICMP패킷에 포트 번호가 기재되어 있는 행은 전부 UDP 포트 도달 불가에 관한 것이기 때문에 목적지 포트 번호 필드가 기재된 ICMP패킷은 모두 UDP로 변경 처리하여 학습하도록 하였다.

프로토콜에 따라 역할을 달리하는 포트 번호 처리 : 동일한 포트번호를 사용하더라도 프로토콜에 따라서 서로 다른 인터넷 응용 프로그램이 있다 [20]. 예컨대, 네트워크 서비스를 제공하는 포트 번호는 컴퓨터의 파일 시스템에 보관되므로 일반 사용자가 포트 번호를 직접 지정하는 경우는 없다. 사용자가 연결을 원하는 서버의 호스트 IP 주소만 클라이언트 프로그램에게 지정하고, 포트 번호 선택은 프로그램에서 자동으로 해준다. TCP와 UDP는 별도의 포트 주소 공간을 관리하므로 동일한 포트 번호를 사용할 수 있다. 21, 23, 80 번 등의 포트는 FTP, Telnet, HTTP로 약속이 되어 있고 동일 포트로 사용이 가능하다. 이와 같이 활용될 프로토콜에 따라 역할을 달리하는 포트 번호는 Word2Vec 기법에서 동음이의어로 학습되어 성능 저하를 유발할 가능성이 있다. 따라서 본고에서는 이를 방지하기 위해서 프로토콜과 수신지 포트번호를 결합하여 '프로토콜_포트번호(수신지)'의 형태로 한 개의 단어로 동작하도록 하였다. 표 1과 표 2에서 제안기법의 4-tuple 전처리의 예시를 보여준다.

(표 1) 전처리 전의 데이터
(Table 1) Data before the processing

	프로토콜	송신지 IP주소	수신지 IP주소	수신지 포트번호
1	TCP	147.32.84.9	147.32.69.7	53
2	UDP	147.32.84.32	147.32.81.1	53
3	ICMP	147.32.84.11	147.32.73.17	7777

(표 2) 전처리 후의 데이터
(Table 2) Data after the processing

	프로토콜	송신지 IP주소	수신지 IP주소	프로토콜_수신지 포트번호
1	TCP	147.32.84.9	147.32.69.7	tcp_53
2	UDP	147.32.84.32	147.32.81.1	tcp_53
3	UDP	147.32.84.11	147.32.73.17	udp_777

4. Word2Vec 모델

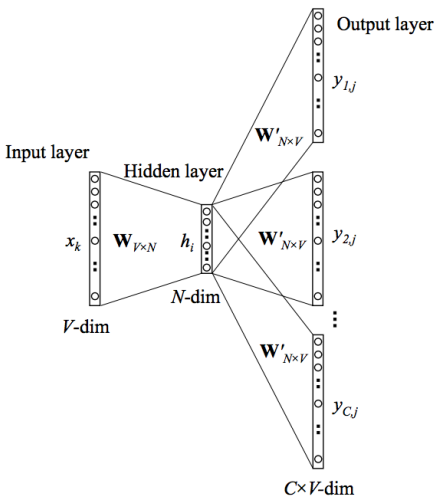
본 장에서는 Word2Vec 모델링 과정과 차원 축소 및 시각화에 대하여 상세히 기술한다.

4.1 개요

Word2Vec 알고리즘은 가공하지 않은 텍스트로부터 학습한 단어 임베딩(embedding)에 대해 계산하는 효율적인 예측 모델이다. Word2Vec은 CBOW(Continuos Bag of Word)와 Skip-gram 모델로 나뉜다 [21]. Word2Vec모델에서 CBOW는 주변 단어(surrounding word)들로부터 타깃 단어(context word)들을 예측하는 반면, Skip-gram은 타깃 단어들로부터 원본 문장의 단어들을 역으로 예측한다는 점을 제외하고 유사하다. CBOW는 주변에 있는 단어들을 가지고 중심에 있는 단어를 맞추는 방식이고, 후자는 중심에 있는 단어로 주변 단어를 예측하는 방법이다. 예컨대, '나는 __에 간다'와 같은 문장을 가정하자. 빈칸에 들어갈 수 있는 단어는 다양하다. '학교'일 수도, '집'일 수도 있다. 이렇듯 주변 단어를 가지고 중심에 있는 타깃 단어를 맞춤으로써 단어 벡터들을 만들어 내는 방법이 CBOW이다. 반대로 '__외나무다리__'와 같은 문장에서 앞뒤로 어떤 단어가 올지 예측하는 방법은 Skip-gram이다. '외나무다리'앞에 '-는'이 올 수 있으며 '-는'앞에는 '원수'가 올 수 있다. Word2Vec알고리즘은 '외나무다리'가 '원수', '-는'과 어떤 연관이 있음을 학습하고 이를 고려하여 단어를 벡터로 만들게 된다.

4.2 Skip-gram 모델 적용

본고에서는 Skip-gram 모델을 적용한다. 그림 1에서 Skip-gram의 아키텍처를 도식하였다. Word2Vec은 입력층(hidden layer), 은닉층(hidden layer), 출력층(output layer)로



(그림 1) Word2Vec의 Skip-gram 아키텍처 (이미지는 [22]에서 발췌하였다.)

(Figure 1) The Skip-gram architecture of Word2Vec (Image taken from [22])

구성되며 계층 및 연결가중치 벡터의 크기는 다음과 같다.

1) Input layer의 크기 = $1 \times V$
2) Hidden layer 크기 = N
3) Input hidden weight의 크기 = $V \times N$
4) Hidden-Output weight의 크기 = $N \times V$
5) Output Layer 크기 = $C(1 \times V)$

Skip-gram 모델에서는 먼저 주변 단어들과 타깃 단어의 관계를 도치하고, 이들 타깃 단어로부터 각 주변 단어의 예측을 시도한다. Word2Vec의 Skip-gram은 아래 수식 (1)을 최대화하는 방향으로 학습을 진행한다.

$$P(o|c) = \frac{\exp(u_o^T v_c)}{\sum_{w=1}^{W} \exp(u_w^T v_c)} \quad (1)$$

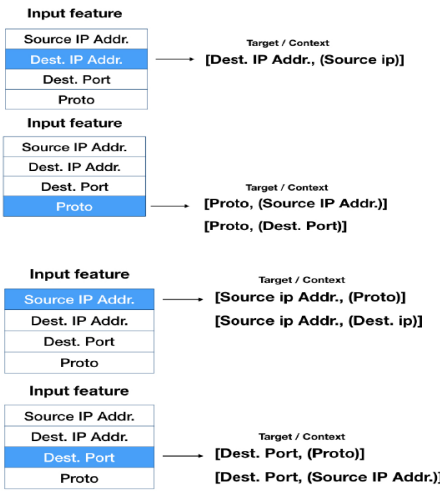
여기서 o는 주변 단어, c는 타깃 단어, T는 단어 최소 빈도수를 의미한다. $p(o|c)$는 타깃 단어가 주어졌을 때 주변단어가 등장할 조건부확률을 의미한다. 수식 (1)을 최대화 한다는 의미는 아래 식 좌변은 타깃 단어가 주어졌을 때 주변 단어가 나타날 확률을 최대화 함을 의미한다. 4.1절의 예에서는 '외나무다리'가 등장했을 때 '원수'라는 표현이 높은 확률로 나오는 모델을 학습하는 것이다. v는 Word2Vec 인공신경망에서 입력층과 은닉층을 연결하는 가중치 행렬 W의 행벡터, u는 Word2Vec 인공신경망에서 은닉층과 출력층을 연결하는 가중치 행렬 W'의 열벡터이다. 우변 분자의 지수의 증가는 타깃 단어에 해당하는 벡터와 주변 단어에 해당하는 벡터의 내적 값의 증가를 의미하며, 벡터 내적은 코사인이므로 내적 값의 증가는 단어 벡터 간 유사도의 증가를 의미한다. 반대로 분모의 감소는 윈도우 크기 내에 등장하지 않는 단어들은 중심 단어와의 유사도를 감소시킨다 [2]. 즉, 하나의 단어가 주어지면, 나머지 주변 단어들의 등장에 대한 확률을 유추하는데 활용할 수 있는데 본고에서는 이를 봇넷 탐지에 적용한 것이다. CBOW 모델과 비교하여 윈도우의 크기에 따라 학습량이 기하급수적으로 증가하기 때문에 통상 CBOW보다 성능이 우수하다고 알려져 있다 [22].

학습을 위해 입력된 단어는 원 핫 벡터(one-hot vector)의 형태로 입력되며, 학습 후 임베딩 된 단어는 벡터 값을 갖는다. 입력층과 출력층은 같은 크기의 차원을 가지며, 은닉층의 뉴런은 입력층보다 저차원이어야 한다. 은닉층에 처음에 설정된 임의의 초기 가중치 값들은 학습이 진행되면서 loss가 최소가 되는 최적의 가중치를 갖도록 학습된다. 결론적으로, Skip-gram 인공신경망의 은닉층에 가중치 행렬 W, W'을 구하는 것이 궁극적인 목표가 된다. 은닉층에는 각 단어의 임베딩 최종 결과가 저장되기 때문에 은닉층이 곧 Word2Vec이라고 할 수 있다.

4.3 비용함수

모든 훈련 과정들에서, 문장(context)에 있는 단어에 포함된 모든 단어에 대한 확률을 계산하고 정규화해야 한다. 따라서 계산 비용이 매우 크다. 계산 비용을 최소화하기 위하여 본 연구에서는 비용함수로 NCE (Noise Contrastive estimation)를 [23] 사용하였다. NCE는 CBOW와 Skip-gram 모델에 사용하는 비용 계산 알고리즘이다. 일반적으로 언어 모델링 또는 단어 임베딩 생성 작업에서 계산적으로 큰 비용이 드는 단계를 피하기 위한 전략으로 언어 모델 추정 문제를 실제 분포에서 얻은 샘플과 인공적으로 만든 노이즈 샘플(noise sample)을 구별하는 이진 분류 문제로 제한해 준다. 실제 단어 쌍뿐만 아니라 허위(noise) 쌍도 같이 만든 후 학습하여 실제 단어에는 높은 확률을 부여하고, noise 단어에는 낮은 확률을 부여함으로써 실제와 noise를 구별한다. 전체 단어 V를 계산하는 것이 아니라, 선택한 N개의 노이즈 단어들에 한정

하여 계산하므로 NCE 비용함수를 사용하면 성능 향상과 더불어 학습시간을 줄이는 효과를 기대할 수 있다.



(그림 2) 제안기법의 Word2Vec모델의 특징 구성
(Figure 2) Relationships among features in our Word2Vec model

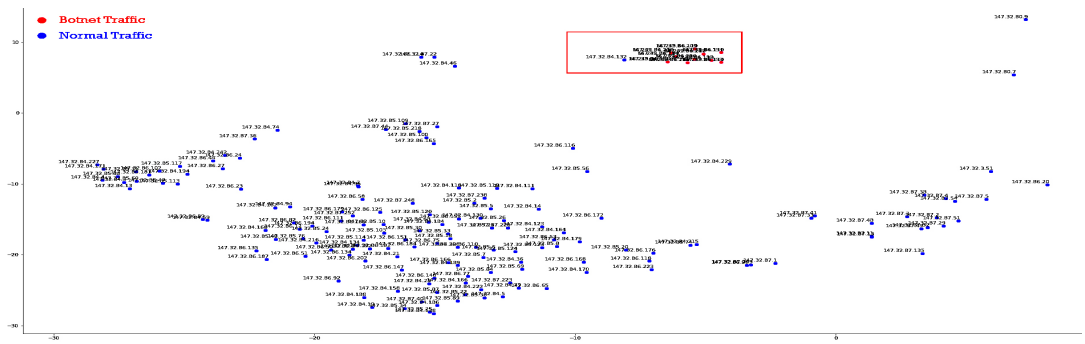
4.4 학습절차

Word2Vec은 자연어를 학습하기 위하여 만든 모델로써 비슷한 맥락에 등장하는 단어들은 유사한 의미를 지니는 경향이 있다는 이론인 Distributional Hypnosis에 [24] 기반을 두고 있기 때문에 윈도우 크기에 따라 비슷한 구간에서 출현하는 자연어끼리의 유사도를 학습하는 모델이므로 한 개의 네트워크 플로우를 한 개의 자연어 문장으로 취급한다. 하지만 윈도우 크기에 종속되어 단어끼리의 학습을 진행하면 연관성이 적은 특징들끼리의 단어 쌍을 구성하는 것은 결과적으로 정확도를 저해한다고 판단하여 본 연구에서는 입력 특징을 기준으로 통신상에 연관이 깊은 특징끼리만 학습하게 하였다. 최종적으로 단어 임베딩의 대표적인 모델인 Word2Vec Skip-gram 방법론에 따라 학습 모델 생성을 진행하였다. 타깃 단어를 기반으로 문장(네트워크 플로우)을 학습하였다. 3.2절에서 서술한 바와 같이 제안하는 Word2Vec모델에서의 단어벡터 구성을 그림 2에 도식하였다.

① 송신지 IP주소가 프로토콜 또는 수신지 IP주소와 단어 쌍을 이룰 수 있다.
② 수신지 IP주소는 목적지 IP주소와 쌍을 이룰 수 있다.
③ 수신지 포트번호는 프로토콜 또는 송신지 IP주소와 쌍을 이룰 수 있다.
④ 프로토콜은 송신지 IP주소 또는 수신지 포트번호와 쌍을 이룰 수 있다.

4.5 차원축소 및 시각화 : t-SNE 적용

본 연구의 CTU-13 데이터셋에서 10번 시나리오를 활용한 27개의 특징 중 넷플로우 4-tuple 전처리를 진행하였다. 전처리된 Word2Vec 인공신경망 Skip-gram 알고리즘을 사용한 벡터 임베딩 모델에 대한 모델 벡터의 상관관계는 고차원 데이터를 2차원으로 줄여 시각적으로 데이터 유사성를 확인하고 제안기법의 클러스터 형성을 직관적으로 확인하기 위하여 t-SNE(t-Stochastic Nearest Neighb



(그림 3) t-SNE 시각화 적용결과 (CTU-13 10번시나리오)
(Figure 3) t-SNE visualization result (10th scenario of CTU-13 dataset)

or) 시각화를 [25] 진행하였다. 그 결과는 그림 3과 같다. 그림 3에서 빨간 점은 봇넷 트래픽을 의미하며, 파란 점은 정상 트래픽을 의미한다. 몇 개의 클러스터를 형성한 것으로 판단된다. 그 중 하나의 클러스터는 봇넷 트래픽을 대부분 포함하고 있다 (그림 3에서 빨간 직사각형 안에 속에 있는 트래픽을 의미한다.).

5. 성능평가

4장에서 설계한 Word2Vec모델 Skip-gram 알고리즘을 이용한 멀웨어 탐지규칙 시스템을 개발하고, 실험을 진행하였다.

5.1 성능평가 환경 및 평가척도

5.1.1 성능 평가 환경

실험에 사용된 연구 개발 환경은 머신러닝 플랫폼 구글 코랩을 활용하였다. Python3.6기반의 Jupyter Notebook을 사용하였으며 Tensorflow 1.13.1버전에서 성능 평가를 진행하였다. 학습을 위해 사용한 시스템 환경은 Intel(R) Xeon(R) CPU@2.30GHz, OS Platform 64비트 Ubuntu 18.04.2 LTS, 13GB RAM, K80 GPU(GPU 11.4GB) 시스템에서 진행되었다.

5.1.2 성능 평가 척도

본고에서는 성능 평가 척도(performance metric)로서 통상적으로 활용되는 분류 정확도(overall accuracy), 정밀도(precision), 재현율(recall), F점수(F-measure)를 활용한다. 성능 평가 척도들은 통계적으로 다음과 같이 정의된다*.

$$\text{분류정확도} = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

$$\text{정밀도} = \frac{TP}{TP+FP}, \ \text{재현율} = \frac{TP}{TP+FN} \quad (3)$$

$$F\text{점수} = \frac{2\times \text{정밀도}\times \text{재현율}}{\text{정밀도}+\text{재현율}} \quad (4)$$

5.1.3 성능 평가 결과

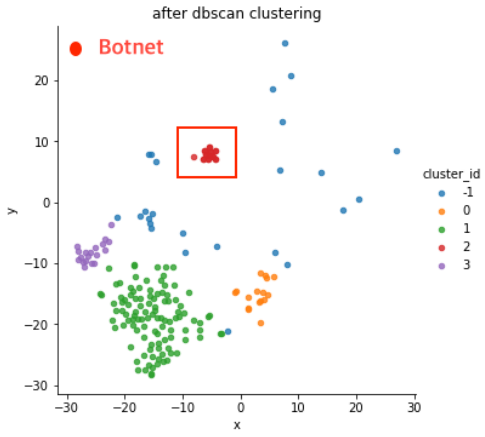
우리는 제안하는 Word2Vec Skim-gram 모델의 봇넷 탐지 성능에 관한 평가를 실험적으로 진행하였으며, 그 결과는 다음과 같다. 분류 정확도 값은 0.997975, 정밀도 값은 1.0, 재현율 값은 0.997933, F점수값은 0.998965를 도출하였다. 이는 마이크로 평균값이다. 본고에서는 마이크로와 매크로의 평균을 각각 따로 도출하였는데, 그 이유는 매크로(Macro) 평균은 각 클래스에 대해 독립적으로 메트릭을 계산한 다음 평균을 취하므로 모든 클래스를 균등하게 처리하지만 마이크로(Micro) 평균은 모든 클래스의 기여를 집계하여 평균 메트릭을 계산한다. 따라서, 다중 클래스 분류 설정에서 클래스 불균형이 의심되는 경우 마이크로(Micro) 평균이 더 바람직하다.

[4]에서 91.02%의 분류정확도와 82.07%의 검출율(Detection rate), 정밀도(precision) 99.02%을 나타내었다. 본 연구는 [4] 에서의 연구 결과에 비해 결과론적으로는 비교 분류에 있어서 4-tuple기반으로 한 멀웨어 단어 벡터들을 분류하고 10회 이상 출현한 멀웨어(Malware)만 정답지를 가지고 트레이닝 데이터셋(Train dataset)으로 분류를 진행하였기 때문에 [4] 에 비해 분류 측면에서의 높은 멀웨어 탐지율에 대한 정확도와 정밀도를 나타내었다고 볼 수 있지만 단순히 트레이닝, 테스트 데이터셋의 수치값으로 평가를 진행하였다는 점과 과적합에 대한 한계점은 가지고 있다.

그림 4에서 DBScan(Density Based Spatial Clustering of Application with noise) 클러스터링 알고리즘을 적용한 결과를 도식하였다. DBScan알고리즘은 군집 간의 거리를 이용하는 K-means 알고리즘과 달리 데이터들의 밀도를 이용한 클러스터링 방식이다. DBScan은 미리 클러스터의 개수를 설정하지 않아도 된다는 장점이 있으며 클러스터의 밀도에 따라서 클러스터를 자동적으로 형성시키기 때문에 기하학적인 모양을 갖는 군집도 잘 찾아낼 수 있다. DBScan 클러스터링은 비정상 트래픽을 더욱 명확하게 분류하기 위한 기법 군집의 개수를 따로 정하지 않아도 비선형 경계의 군집을 구할 수 있기 때문에 Malware(멀웨어) 탐지에 있어서 DBScan클러스터링이 활용된 사례가 다수 있다. 그러나 DBScan 알고리즘은 밀도 위주로 계산이 이루어지기 때문에 높은 계산량을 요구할 수 있다. 본고에서는 1.3M개 정도의 플로우에 대하여 5.1.1에서 기술한 컴퓨팅 시스템을 활용하였는데 학습에 약 8시간 정도 소요되었다. 실제의 활용에 있어서 DBScan 알고리즘을 적용하고자 한다면 수천~수억건의 플로우 데이터를 실시간으로 처리해야 한다. 따라서 DBScan방법을 실시간으로 실제의 봇넷 탐지에 적용하기에는 한계가 있을 것으로 판단한다. 다만, 전체 데이터를 랜덤하게 분할하여 클러스터링

* TP는 true positive의 개수, TN은 true negative의 개수 FP는 false positive의 개수, FN은 false negative의 개수를 의미한다.

을 수행하고 그 결과값을 누적시키는 분할정복 등의 기법으로 계산수행시간을 완화할 수 있으나 여전히 국부 최적화 등의 이슈로부터 자유롭지 못할 것으로 판단된다.



(그림 4) 멀웨어 탐지 디비스캔 클러스터링
(Figure 4) DBScan of Clustering Mallware Detection

6. 결론 및 향후 과제

본 연구에서는 단어 임베딩(Word embeddings)에 대해서 계산하는 효율적인 예측 모델 Word2Vec을 Skip-gram을 활용한 인공신경망 모델 봇넷 탐지(Botnet Detection) 방법 기술을 제안하였다. 해당 Word2Vec 모델을 이용하여 멀웨어 탐지를 위한 전처리 과정과 t-SNE 시각화 및 밀도 기반 클러스터링(Density Based Spatial Clustering of Application with noise)을 적용하였으며, 제안 기법의 우수한 성능을 실험적으로 확인하였다.

향후 본 연구팀에서는 정적인 Netflow 데이터셋 뿐만 아니라 동적인 실시간 데이터에 적용할 수 있도록 제안 기법을 확장할 계획이다.

참고문헌(Reference)

[1] Vasiliadis et al, “MIDeA: a multi-parallel intrusion detection architecture,” In ACM conference on Computer and communications security (CCS) 2011. https://dl.acm.org/citation.cfm?id=2046741

[2] Song Yangqiu et al, “Unsupervised sparse vector densification for short text similarity,” Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2015. https://aclweb.org/anthology/N15-1138

[3] M. Tomas et al. “Distributed representations of words and phrases and their compositionality.” Advances in Neural Information Processing Systems (NIPS) 2013. https://dl.acm.org/citation.cfm?id=2999959

[4] R. S. M. Carrasco et al, “Unsupervised intrusion detection through skip-gram models of network behavior.” Computers & Security 78 (2018): 187-197. https://doi.org/10.1016/j.cose.2018.07.003

[5] Popov, I. “Malware detection using machine learning based on Word2Vec embeddings of machine code instructions” Siberian Symposium on Data Science and Engineering 2017. https://ieeexplore.ieee.org/document/8071952

[6] S. Garcia, M. Grill, “An empirical comparison of botnet detection methods,” Computers & Security, vol. 45, pp. 100 - 123, 2014. https://doi.org/10.1016/j.cose.2014.05.011

[7] L. Maaten et al, “Visualizing data using t-SNE.” Journal of machine learning research 9.Nov (2008): 2579-2605. http://www.jmlr.org/papers/v9/vandermaaten08a.html

[8] E. Martin et al. “A density-based algorithm for discovering clusters in large spatial databases with noise,” Kdd. Vol. 96. No. 34. 1996. https://dl.acm.org/citation.cfm?id=3001507

[9] S. Frank et al. “Feature engineering in context-dependent deep neural networks for conversational speech transcription,” 2011 IEEE Workshop on Automatic Speech Recognition & Understanding, 2011. https://doi.org/10.1109/ASRU.2011.6163899

[10] S. Lee et al, “LARGen: Automatic Signature Generation for Malwares Using Latent Dirichlet Allocation,” IEEE Transactions on Dependable and Secure Computing (TDSC) Vol.15(5), pp. 771- 783, 2018. https://doi.org/10.1109/TDSC.2016.2609907

[11] G. Salton et al, “A Vector space model for automatic indexing.” Communications of the ACM, Vol.18(11), pp. 613-620, 1975. https://doi.org/10.1145/361219.361220

[12] D. Scott, et al. “Indexing by latent semantic analysis,” Journal of the American society for information science 41.6, 391-407, 1990.

https://doi.org/10.1002/(SICI)1097-4571(199009)41:6<391::AID-ASI1>3.0.CO;2-9

[13] H. Thomas. “Probabilistic latent semantic analysis,” Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence. Morgan Kaufmann Publishers Inc., 1999. https://dl.acm.org/citation.cfm?id=2073829

[14] T. N. Rubin et al, “Statistical topic models for multi-label document classification,” Machine Learning, Vol.88 (1-2), pp. 157-208, 2012. https://doi.org/10.1007/s10994-011-5272-5

[15] H. G. Kim et al. “Visualization of Malwares for Classification Through Deep Learning,” Journal of Internet Computing and Services(JICS), 19(5), pp. 67-75, Oct. 2018. http://dx.doi.org/10.7472/jksii.2018.19.5.67

[16] E. Hodo et al, “Shallow and deep networks intrusion detection system: A taxonomy and survey”, arXiv preprint arXiv:1701.02145 https://arxiv.org/abs/1701.02145

[17] S. Ryu et al. A Comparative Study of Machine Learning Algorithms and Their Ensembles for Botnet Detection. Journal of Computer and Communications, 6(5), 119-129, 2018. https://dx.doi.org/10.4236/jcc.2018.65010

[18] S. Lee et al., “NeTraMark: a network traffic classification benchmark,” ACM SIGCOMM Computer Communication Review 41.1, 22-30, 2011. http://doi.acm.org/10.1145/1925861.1925865

[19] K. C. Claffy et al, “A parameterizable methodology for Internet traffic flow profiling.” IEEE Journal on selected areas in communications, 13.8, 1481-1494, 1995. https://doi.org/10.1109/49.464717

[20] P. Sethi et al, “Internet of things: architectures, protocols, and applications,” Journal of Electrical and Computer Engineering, 2017. https://doi.org/10.1155/2017/9324035

[21] L. Yang et al. “Topical word embeddings,” Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.703.7444&rep=rep1&type=pdf

[22] X. Rong, “Word2Vec parameter learning explained,” arXiv:1411.2738, 2014. https://arxiv.org/abs/1411.2738

[23] Mnih, Andriy, and Koray Kavukcuoglu. “Learning word embeddings efficiently with noise-contrastive estimation.” Advances in Neural Information Processing Systems, 2013. (NIPS 2013) http://papers.nips.cc/paper/5165-learning-word-embeddings-efficiently-with

[24] A. Kilgarriff. “Thesauruses for natural language processing.” International Conference on Natural Language Processing and Knowledge Engineering, 2003. Proceedings, IEEE, 2003. https://doi.org/10.1109/NLPKE.2003.1275859

[25] L. Maaten et al, “Visualizing data using t-SNE.” Journal of machine learning research, 9, 2579-2605, Nov. 2008. http://www.jmlr.org/papers/v9/vandermaaten08a.html

◐ 저 자 소 개 ◑

이 태 일(Taeil-Il Lee)
2013년~현재 한국교통대학교 철도대학 철도경영·물류·컴퓨터학부(컴퓨터정보공학전공)
관심분야 : 정보통신 및 보안, 인공지능
E-mail : timtaeil@gmail.com

김 관 현(Gwan-Hyeon Kim)
2017년~현재 한국교통대학교 철도대학 철도경영·물류·컴퓨터학부(컴퓨터정보공학전공)
관심분야 : 정보보안 및 인공지능
E-mail : kgh940525@gmail.com

이 지 현(Ji-Hyeon Lee)
2017년~현재 한국교통대학교 철도대학 철도경영·물류·컴퓨터학부(컴퓨터정보공학전공)
관심분야 : 정보보안 및 인공지능
E-mail : dlwl0319@gmail.com

이 수 철(Suchul Lee)
2008년 서울대학교 전기·컴퓨터공학부(공학사)
2014년 서울대학교 대학원 컴퓨터공학부(공학박사)
2014년~2016년 한국전자통신연구원 부설연구소 연구원
2016년~현재 한국교통대학교 철도대학 철도경영·물류·컴퓨터학부(컴퓨터정보공학전공) 조교수
관심분야 : 정보통신 및 보안, 인공지능
E-mail : sclee@ut.ac.kr