텐서플로우 기반의 기계학습 보안 프로그램

윤성권, 박태용

호원대학교

Machine-Learning Anti-Virus Program Based on TensorFlow

Seong-kwon Yoon, Tae-yong Park

Howon University

E-mail : ysk37ysk37@naver.com

요 약

최근 북한은 핵실험, 미사일 발사 등 물리적 도발은 물론 고위 공직자에 대한 스마트폰 해킹, 금융권에 대한 디도스(DDoS) 공격 등 사이버 테러를 감행하며 한반도 내 위협의 수위를 높이고 있다. 취약점에 대한 해킹, 악성코드 배포 등을 통해 이루어지는 사이버 공격은 일반적으로 최초의 침입과 공격 징후가 감지된 후 데이터 분석을 통해 백신의 라이브러리 추가 및 업데이트, 보안 취약성을 보완하는 등 소극적인 대응 방법을 취하고 있다. 본 논문에서는 프로그램 스스로 취약점을 분석하고 자가 라이브러리 추가, 보안 취약점 해결 등을 수행하는 구글 텐서플로우(TensorFlow) 기반의 기계학습 능력을 갖춘 보안 프로그램에 관한 개념을 연구하고 제안하였다.

ABSTRACT

Peace on the Korean Peninsula is threatened by physical aggressions and cyber terrors such as nuclear tests, missile launchings, senior government officials' smart phone hackings and DDos attacks to banking systems. Cyber attacks such as vulnerability for the hackings, malware distributions are generally defended by passive defense through the detecting signs of first invasion and attack, data analysis, adding library and updating vaccine programs. In this paper the concept of security program based on Google TensorFlow machine learning ability to perform adding libraries and solving security vulnerabilities by itself is researched and proposed.

키워드

기계학습, 머신러닝, 보안프로그램, 백신, Machine Learning, Security Program, Vaccine

Ⅰ. 서 론

SF영화에서 쉽게 볼 수 있었던 인공지능은 Google Deep Mind에서 개발한 인공지능 알파고(Alpha Go)로 인해 실제의 세상 속으로 다가왔다[1, 2]. 알파고의 대국을 보면서 가까운 미래에는 인공지능이 사람이 하는 일을 대체 할 수 있을 것이라는 예측이 많이 나오고 있다. 그러나 이러한 인공지능이 탄생하기 위해서는 기계학습의 발전이 먼저 이루어져야 한다[3].

본 논문은 구글(Google)의 오픈소스 2세대 머신러닝 시스템인 텐서플로우(Tensorflow)[4]를 기반으로 하여 정보보안 정책수립 및 해킹에 대한 방어를 할 수 있는 '인공지능 프로그램'(이하 프로그램)을 제안하고자 한다. 기계학습 방법 중 지도학습을 통해 인공지능이 독자적으로 악성코드에 대한 분석을 하고 이에 대응 할 보안 정책을 수립하는 것은 불가능에 가깝다. 그러므로 비(非)지도학습을 이용하고, 분석된 악성코드에 대한 최적의 대응 제거코드를 만들기 위한 강화학습[5]을 적용하며 인간의 '기억'이라는 행위를 단계별로 나누어져 구현할 예정이다[5, 6]. 또한 파이썬(Python) 의 'Scikit-learn[7]'을 이용하고, 쌓여진 경험에 대해 데이터 마이닝 과정을 거칠 수 있도록 하여 최적의 값을 도출해 낼 것이다.

그리고 악성코드에 대한 패턴인식 등에 이용하

기 위해 인공신경망을 적용하고[8,9,10], 최적의 값을 구하기 위해 유전 알고리즘을 적용할 것이다[11].

II. 텐서플로 기반 프로그램 구현

2.1. 프로젝트 개요

Google Brain 팀의 첫 머신러닝시스템(Machine Learning System) Deep Learning Infrastructure 인 디스트빌리프(DistBelief)는 2011년 까지 Google 내부에서 사용되었다[12]. 디스트빌리프는 YouTube, Improve Speech Recognition, Google Photo, Automated Image Captioning, Deep Dread 등에서 성공적인 결과를 내며 구글의 서비스와 연구에 사용되었다[13]. 하지만 Deep Learning 플랫폼은 구글의 내부 구조와 밀접하게 연관되어 있어 지금까지 공개되지 않고 있다.

구글은 다양한 서비스로 검증된 디스트빌리프의 단점을 보완한 텐서플로우를 공개하였다[1, 14]. 기계학습은 특정 Task에 대해 경험이 있어야 성능이 올라가는데[15], 경험(Experience)으로 작용할 수 있는 데이터에 대한 부분은 공개 하지 않았다. 그렇기 때문에 임의의 악성코드에 이상적인 대처를 위해서는 많은 양의 경험으로서의 데이터를 찾아야만 한다.

제안하는 프로그램은 리눅스에서 코딩, 구현, 테스트 과정을 거칠 예정이다. 이후 클러스터로 옮겨서 빅 데이터에 대한 학습을 할 예정이며, 클러스터는 다수의 라즈베리파이(Model 3 type B)를 병렬 연결하여 만들어서 사용할 것이다[16].

2.2. 개략적인 알고리즘

제안하는 프로그램은 데이터 습득, 데이터 학습, 데이터 응용 등 3개의 주요 프로세스로 이루어져있다. 인간의 기억작용과 유사한 작동을 위해서 3개의 프로세스와 함께 다음과 같은 시스템이 추가로 구성된다.

표 1. 추가 시스템 항목

구 분	임 무
Memorize System	입력받은 정보를 데이터로서 기록, 저장한다.
Saving System	입력받은 데이터를 보존한다.
Playing System	보존한 정보를 재생이 가능하게 한다.
Recognize System	재생된 정보가 원하는 정보와 일치하는지 확인한다.

'Memorize System'은 최초로 입력받은 텍스트 파일 혹은 소스코드에 대하여 입력받은 모든 파일을 학습에 영향을 미치지 않는 외부저장장치에 임시저장을 하고, 이름, 크기, 일부의 내용에 대하여 내부 장치에 기록 및 저장을 한다. 그 이후 데이터 습득 프로세스는 기록된 파일에서 학습하게 될 파일을 찾아 내부 저장장치로 가지고 온다. 한꺼번에 많은 양의 데이터가 학습 프로세스로 유입되지 않게 컨트롤 할 수 있는 기능 또한 하게 된다.

다음 그림은 데이터 학습 프로세스의 일부로 데이터로서 획득한 악성코드에 대해 Revers Engineering을 수행하고[17,18,19] 소스코드 모델링 및 적절한 데이터 분류 작업 이후 업데이트를 거쳐 결과(P)를 추출해내는 절차를 나타낸 것이다[10,20].

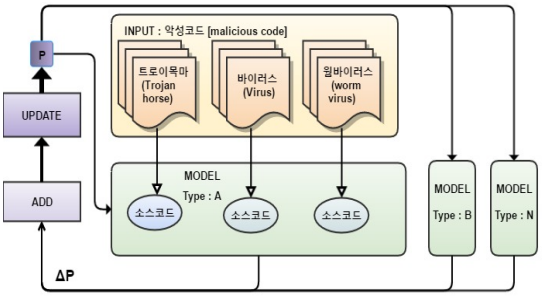

그림 1. Virus_Library training and Updating used Reversing

결과(P)를 획득하게 되면 Saving System을 통해 데이터 습득 과정에 나누어져 기록된 항목 리스트로 분류가 이루어진다. 그림과 같이 소스코드를 모델링 하고 업데이트가 이루어지면 결과(P)는 컴퓨터 바이러스 항목에 들어갈 것이고, 그 내용이 트로이목마와 관련된 바이러스의 분석이었다면 알맞은 세부적인 항목에 결과가 저장 될 것이다.

응용 프로세스에서는 학습프로세스에서 부족한 데이터 마이닝, 핸들링과 유전 알고리즘을 통해 데이터를 한층 더 다듬는다. 그리고 인공신경망[10]을 통해 사용가능한 데이터의 추출이 이루어지며 Recognize System이 이루어진다. 최종적으로 Playing System과 Recognize System이 한 번 더 작용하면서 출력이 이루어지면서 나타나는 작업이 정확한지 검수를 하게 되고, 문제가 없다면 하나의 항목에 대한 학습이 이루어 진 것이다.

응용 프로세스에서 이루어진 유전 알고리즘의 개략적 연산 작업과 인공신경망은 그림2와 같다.

임의의 상황A(type A)의 발생이 이루어졌을 때 이에 맞는 최적의 값을 도출해 내는 방법은, 먼저 기존에 이루어졌던 학습을 통해 지금 일어난 일에 대하여 상황 A로 스스로 정의를 내리는 것으로 시작한다. 그리고 상황 A에 대한 연관 또는 이전 경험을 바탕으로 대응에 알맞은 최적의 값을 추출, 코딩작업을 수행 하게 되는 신경망 모델이다.

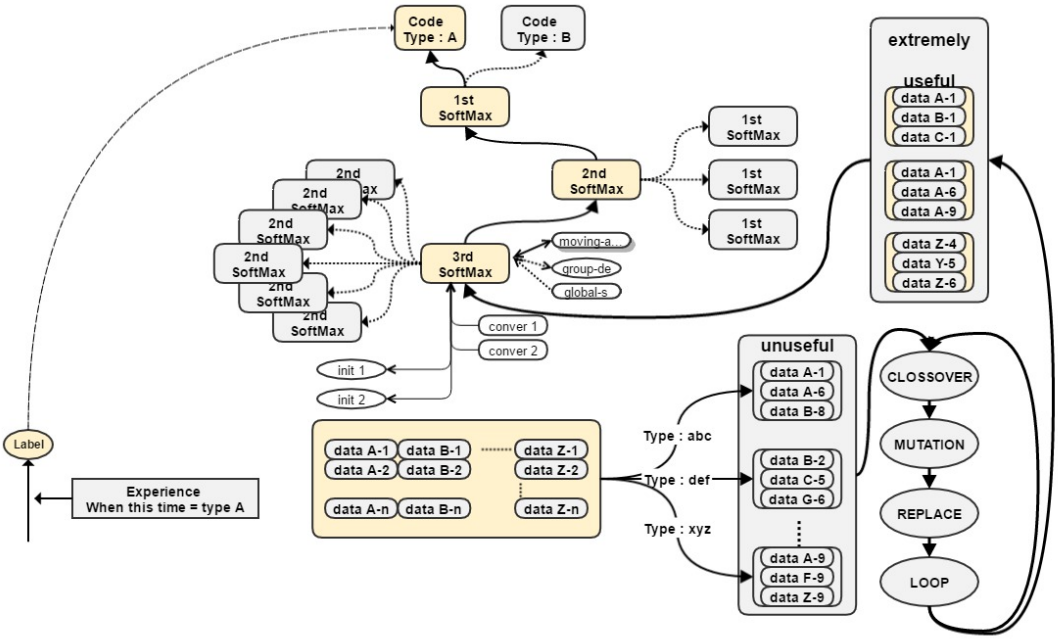

그림 2. Visualization Genetic algorithm along with neural network simple model

2.3. 제안하는 프로그램이 수행하는 업무

알고리즘의 구현이 정상적으로 이루어지고 이를 통해 충분한 학습이 이루어지면 본 논문에서 구현하는 프로그램은 다음과 같은 업무를 수행할 것이다.

·백신서비스(Vaccine service)
현재의 백신은 수동적인 업데이트가 이루어져야 했다. 이후의 백신은 자동화된 업데이트로 시시각각 외부에서의 악의적인 바이러스 감염에 대하여 대처할 수 있을 것이다.

·방화벽(Fire wall)
백신 서비스와 마찬가지로 프로그램은 외부에서의 비인가 침입 등에 대한 훈련을 통해 취약점 대응 방안을 스스로 습득한다. 또한 외부에서 DDos등 악의적인 공격에 대하여 상대를 판별하고 트래픽을 차단하는 기능이 도입될 것이다.

·프로그래밍(Programing)
다수의 악성코드와 소스코드를 분석함으로서 누적된 데이터를 통해 유사한 소스코드를 계속해 생성해 나아갈 수 있고, 생성된 코드를 다시 분석하여 단점을 보완한 프로그래밍 작업을 수행할 것이다.

이와 같이 인간이 직접 해야 했던 작업을 프로그램이 대신하여 수행할 수 있게 될 것이다.

III. 결 론

인공지능을 크게 분류하면 강(强)인공지능과 약(弱)인공지능으로 분류할 수 있다. 현재 약인공지능에 위치해 있는데, 이런 인공지능이 강인공지능이 되기 위해서는 자의식이라는 존재가 필요하다. 그 자의식과 유사하게 스스로 판단을 내릴 수 있고, 새로운 코드를 작성해 나아갈 수 있는 프로그램을 만들고자 하는 것이 논문의 목적이다.

이를 위해 본 논문에서는 텐서플로우를 활용한 기계학습 알고리즘으로 기존의 백신서비스와 달리 자동으로 분석과 업데이트를 수행하도록 하는 방안을 제안하였다. 원활한 기계학습을 위해서는 충분한 경험에 알맞은 데이터가 갖추어져야하며 [15], 보안프로그램에 대한 기계학습 영역을 보강하여야한다. 향후 충분한 데이터와 실험을 통해 본 논문에서 제안한 프로세스를 보완하여 기계학습 보안프로그램을 완성할 예정이다.

참고문헌

[1] Mohammadreza Asghari Oskoei, Artificial intelligence a guide to intelligent systems, 2012
[2] David Silver et al., “Mastering the game of Go with deep neural networks and tree search. Nature,” (529):484–489, 2016.
[3] A. L. Tarca et al., “Machine learning and

its applications to biology," PLoS Comput. Biol., vol. 3, no. 6, p.116, June, 2007.
[4] Martin Abadi et al. "TensorFlow_Large-Scale Machine Learning on Heterogeneous Distributed Systems", (Cornell Univ), November 9, 2015
[5] Arun Nair, et al. "Massively parallel methods for deep reinforcement learning." arXiv preprint arXiv:1507.04296, 2015. arxiv.org/abs/1507.04296.
[6] 김주훈 외 4명. "ART와 퍼지를 혼합한 인간의 기억 모델", 한국지능시스템학회 학술발표 논문집 14(2), 2004.10, 289-292 (4 pages)
[7] F. Pedregosa et al., "Scikit-learn: Machine learning in python," J. Mach. Learn. Res., vol. 12, Nov. 2011.
[8] R. Collobert and J. Weston. A unified architecture for natural language processing: deep neural networks with multitask learning. In ICML, 2008. http://dl.acm.org/ citation.cfm?id=1390177
[9] Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531. 451
[10] Georg Heigold et al. "Multilingual acoustic models using distributed deep neural networks," In Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, pages 8619–8623. IEEE, 2013. research. google.com/pubs/archive/40807.pdf.
[11] K. Deb, et al.,"A fast and elitist multiobjective genetic algorithm: NSGA-II",Apr 2002 http://ieeexplore.ieee.org/xpl/abstractReferences.jsp?tp=&arnumber=996017&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D996017
[12] Jeffrey Dean et al. Large scale distributed deep networks. In NIPS, 2012. Google Research PDF.
[13] http://googleresearch.blogspot.kr/2015/11/tensorflow-googles-latest-machine_9.html, November 09, 2015
[14] https://www.tensorflow.org/versions/master/ get_started/index.html#introduction
[15] Tom M. Mitchell, "Machine Learning", Carnegie Mellon University, April 30th 1986
[16] Simon J. Cox, James T. Cox, Richard P. Boardman, Steven J. Johnston, Mark Scott, Neil S. O'Brien, Iridis-pi: a low-cost, compact demonstration cluster, Cluster Computing, June 2013. DOI: 10.1007/s10586-013-0282-7 Full text: http://link.springer.com/article/10.1007%2Fs10586-013-0282-7
[17] Anton Crombach, et al. ,"Efficient reverse-engineering of a developmental gene regulatory network", July 12, 2012
[18] Reverse engineering gene networks using singular value decomposition and robust regression
[19] 온기원, 이범식 외 3명. "병렬 프로그램 디버깅 기술," 한국전자통신연구소 정보과학지 제14권 6호, 85('96.6) pp.63-72
[20] Alex Krizhevsky, "One weird trick for parallelizing convolutional neural networks" April 29, 2014