International Journal of Advanced Culture Technology Vol.9 No.3 291-297 (2021)
DOI https://doi.org/10.17703/IJACT.2021.9.3.291

IJACT 21-9-38

Analysis of Deep Learning Methods for Classification and Detection of Malware

Phil-Joo Moon

Professor, Dept. of Information & Communications, Pyeongtaek University, Korea
pjmoon@ptu.ac.kr

Abstract

Recently, as the number of new and variant malicious codes has increased exponentially, malware warnings are being issued to PC and smartphone users. Malware is becoming more and more intelligent. Efforts to protect personal information are becoming more and more important as social issues are used to stimulate the interest of PC users and allow users to directly download malicious codes. In this way, it is difficult to prevent malicious code because malicious code infiltrates in various forms. As a countermeasure to solve these problems, many studies are being conducted to apply deep learning. In this paper, we investigate and analyze various deep learning methods to detect and classify malware.

Keywords: *Deep Learning, Malware, Classification, Detection*

1. INTRODUCTION

Recently , Due to the teleworking and Internet use are increasing worldwide, resulting in an explosive increase in traffic and an increase in cyberattacks. Malware traffic also accounted for a significant portion of this traffic. As malware becomes more and more intelligent and the methods of dissemination of malware evolve rapidly, it is difficult to prevent malware. To solve this problem, this paper intends to investigate and compare various deep learning methods to classify and detect malware.

2. MALWARE AND DEEP LEARNING METHOD

2.1 Malware and Deep Learning

Malware is a generic term for all software that can adversely affect computers, servers, clients, and computer networks. In the past, only computer viruses were active and propagated along storage media such as disk copying, but as networks have developed, the number of infections through e-mail or the web has increased significantly.[1]

Malware detection refers to the process of detecting the presence of malware on a host system or of distinguishing whether a specific program is malicious or benign.[2]

Malware Classification is the process of assigning a malware sample to a specific malware family. Malware within a family shares similar properties that can be used to create signatures for detection and classification. Signatures can be categorized as static or dynamic based on how they are extracted. A static signature can be based on a byte-code sequence, binary assembly instruction, or an imported Dynamic Link Library (DLL). Dynamic signatures can be based on file system activities, terminal commands, network communications, or function and system call sequences.[3]

Manuscript received: July 31, 2021 / revised: August 25, 2021 / accepted: August 30, 2021
Corresponding Author: pjmoon@ptu.ac.kr
Tel:+82-31-659-8281, Fax: +82-31-659-8011
Professor, Dept. of Information and Communications, Pyeongtaek University, Korea

Deep learning is defined as a set of machine learning algorithms that attempt high-level abstraction through a combination of several nonlinear transformation methods.[4]

2.2 Deep Learning Methods

2.2.1 Autoencoder

Autoencoder is a kind of artificial neural network used to learn efficient coding (unsupervised learning) of unlabeled data. Encodings are validated and materialized by attempting to regenerate the input from the encoding. Autoencoders learn representations (encodings) for a data set by training a network to ignore unimportant data ("noise"), usually for dimensionality reduction.[5] The goal of AutoEncoder is to encode a representation of an input layer into a hidden layer and then decode it into an output layer, producing values that are identical to (or as close as possible to) the input layer. Figure 1 shows the one-layer AutoEncoder model.

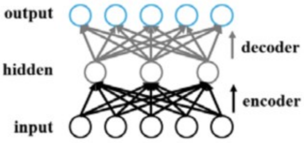

Figure 1. A one-layer AutoEncoder model

2.2.2 CNN

CNN(Convolutional Neural Network) consists of one or several convolutional layers and general artificial neural network layers on top of it, and additionally utilizes weights and pooling layers. Thanks to this structure, CNN can fully utilize the input data of the two-dimensional structure. Compared with other deep learning structures, CNN shows good performance in both video and audio fields.[6] Figure 2 shows the CNN architecture.

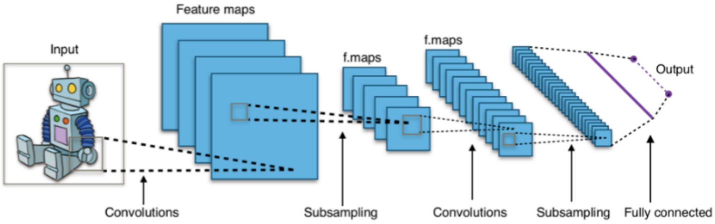

Figure 2. CNN Architecture

2.2.3 RNN

RNN(Recurrent Neural Network) refers to a neural network in which connections between units constituting an artificial neural network constitute a directed cycle. Unlike forward neural networks, recurrent neural networks can utilize the memory inside the neural network to process arbitrary inputs. Due to these characteristics, the recurrent neural network is being used in fields such as handwriting recognition and shows a high recognition rate.[7] Figure 3 shows the RNN architecture.

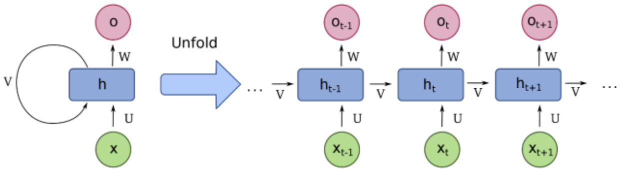

Figure 3. RNN Architecture

2.2.4 RBM

In the Boltzmann machine, it is a model with no interlayer connection. If the interlayer connection is removed, the machine becomes in the form of an undirected bipartite graph composed of Visible Units and Hidden Units. In conclusion, the neural network can be deepened with the benefit of eliminating the layer-to-layer connections of the model.[8] Figure 4 shows the RBM architecture.

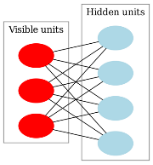

Figure 4. RBM Architecture

3. DEEP LEARNING METHODS FOR MALWARE CLASSIFICATION AND DETECTION

3.1 Wang and Yiu[9]

Wang and Yiu built a machine learning (ML) based malware classifier based on a sequence of API calls to perform malware detection or classification. An RNN-based autoencoder (RNN-AE) can automatically learn a low-dimensional representation of malicious code from a sequence of raw API calls. Multiple decoders can be trained under different supervision to provide more information than the class or family label of the malware.

Figure 5 shows the configuration of the Multi-task Malware Learning Model. This model consists of two decoders. One is for classifying malware and the other is for generating file access patterns (FAPs) that take into account the sequence of API calls of malware.

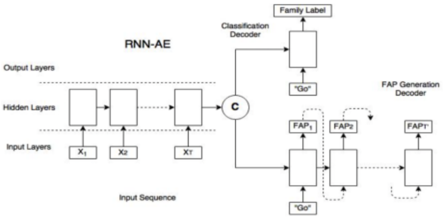

Figure 5. Multi-task Malware Learning Model

3.2 David and Netanyahu[10]

David and Netanyahu use a Deep Belief Network (DBN) implemented as a deep stack of denoising autoencoders to generate an immutable, succinct representation of malware behavior. DBN-generated signatures can accurately classify new malware variants. This method can successfully train deep neural networks used to generate malware signatures using raw input from the sandbox.

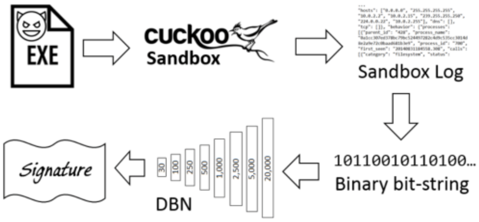

Figure 6. Illustration of all the stages from initial malware run in Sandbox to signature derivation using DBN

Figure 6 shows an end-to-end method configured for automatic signature generation. The program runs in a sandbox, and the sandbox file is converted into a string of binary bits fed to the neural network. The deep neural network generates a vector of size 30 in the output layer, which is treated as the signature of the program.

3.3 Hardy et al.[11]

Hardy et al. created a SAE(stacked AutoEncoders) model with AutoEncoders connected in a daisy chain to form a hierarchical stack. They adds a classifier to the top layer to use SAE for malware detection. Figure 7 shows the model composed of SAEs and a classifier.

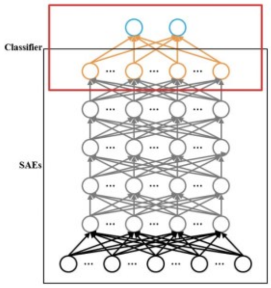

Figure 7. Deep Learning Model for Malware Detection

3.4 McLaughlin et al.[12]

McLaughlin et al. propose a malware detection method that uses a convolutional network to process the raw Dalvik bytecode of an Android application. Figure 8 shows the overall structure of the malware detection network. Android applications are disassembled into raw Dalvik bytecode sequences, which are processed in a convolutional network.

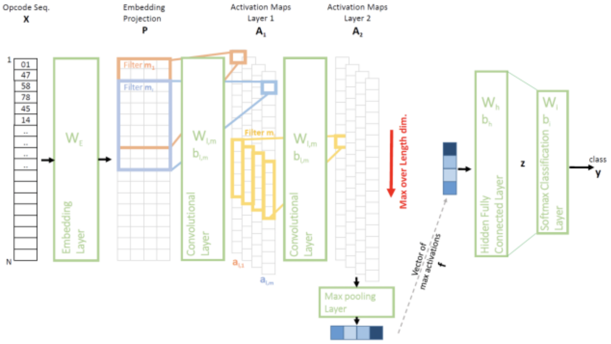

Figure 8. Malware Detection Network Architecture

3.5 Kolosnjaji et al.[13]

Kolosnjaji et al. combine convolutional and recursive layers in one neural network. Figure 9 shows the architecture of a deep neural network. The convolutional part consists of convolutional and pooling layers. On the one hand, the convolutional layer is responsible for extracting features from the raw one-hot vector. Convolution captures the correlation between adjacent input vectors and creates new features. After each convolutional layer, we use max pooling to double the dimensions of our data. The output of the convolutional part of the neural network is connected to the recursive part. Pass each output of the convolution filter as a vector. The resulting sequence is modeled using LSTM cells. A recursive layer can be used to explicitly model sequential dependencies in kernel API traces.

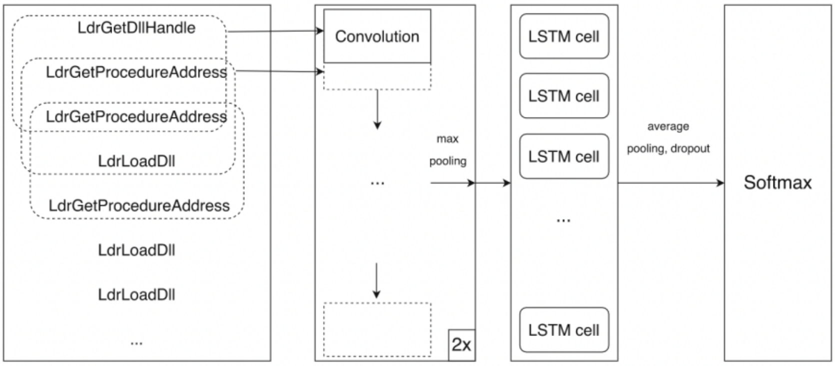

Figure 9. Deep neural network architecture of Kolosnjaji et al.

3.6 Shibahara et al.[14]

Malicious communication detection uses network-based signatures, which are defined as invariant patterns of certain types of malicious communication. Network-based signatures are extracted from malware communications collected by dynamic analysis where malware samples are run. Figure 10 shows an overview of the proposed method.

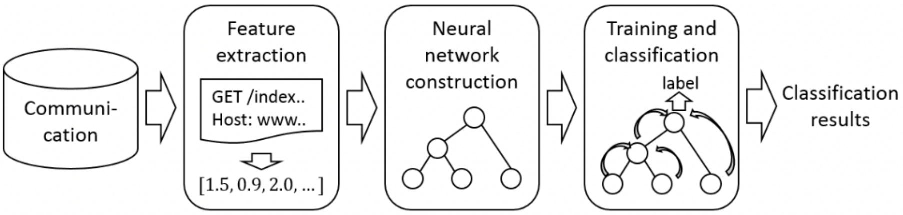

Figure 10. Overview of Proposed Method

3.7 Yuan et al.[15]

As shown in Figure 11, the construction of the deep learning model consists of two stages: an unsupervised pre-training stage and a supervised backpropagation stage. In the pre-training phase, the DBN is built hierarchically by stacking multiple Restricted Boltzmann Machines (RBMs). In the backpropagation step, the pretrained DBN is fine-tuned with labeled samples in a supervised manner.

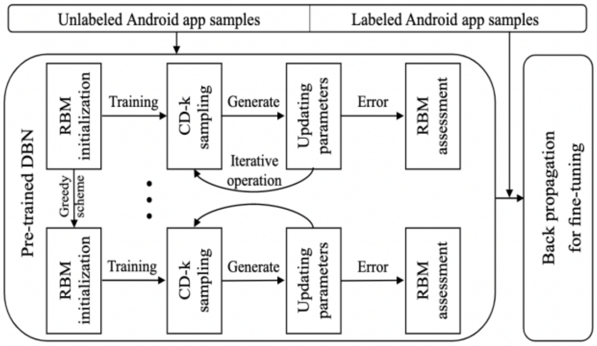

Figure 11. Deep learning model constructed with DBN

4. ANALYSIS OF DEEP LEARNING METHODS TOOLS FOR MALWARE CLASSIFICATION AND DETECTION

This section compares deep learning methods for the classification and detection of malware. Table 1 compares each of datasets, features, malware classification or detection, and malware execution environment according to deep learning methods.

It can be seen that most autoencoders are used for malware classification, and various deep learning methods such as CNN, RNN, and RBM are used for malware detection. Malware detection refers to the process of detecting the presence of malware on a host system or distinguishing whether a particular program is malicious or benign, Malware classification refers to the process of assigning malware samples to specific malware families.

Table 1. Comparison of DL Approaches for Malware Classification and Detection

DL Method	Citation	Dataset	Feature	Security Application	Malware Environment
Autoencoder	Wang and Yiu	API call sequences	PE header information	Malware Classification	
Autoencoder	David and Netanyahu	API call sequences	PE header information	Malware Classification	Cuckoo sandbox
Autoencoder	Hardy et al.	API call sequences	PE header information	Malware Detection	
CNN	McLaughlin et al.	android app	-required permission -sensitive API -dynamic behavior	Malware Detection	Android phone
CNN RNN	Kolosnjaji et al.	system call sequences	PE header information	Malware Detection	Cuckoo sandbox
RNN	Shibahara et al.	network behavior		Malware Detection	
RBM	Yuan et al.	android app	-required permission -sensitive API -dynamic behavior	Malware Detection	Android phone

Dataset uses API call, system call, network behavior, etc. The system call is an interface for accessing the kernel according to the request of the application for the service provided by the kernel of the operating system. An interface that allows you to control the functions provided by an operating system or programming language for use. API call and system call are data sets collected within the host, but network behavior is a data set collected between host and host. It can be seen that the deep learning method using the API call and system call dataset classifies or detects malware in the host, and the deep learning method using the network behavior dataset detects malware on the network.

The environment in which the malware runs is a host, Android phone, or Cuckoo sandbox. Cuckoo sandbox is open source software for automating the analysis of suspicious files, monitoring the behavior of malicious processes while the malware runs in an isolated environment.

5. CONCLUSION

This paper reviews deep learning methods and compares the characteristics of deep learning methods to classify and detect malware.

Malware is software designed to interfere with the normal functioning of a computer. A file or code passed over a network that infects, seeks, steals, or performs virtually any action an attacker wants. Also, since malware exists in many different variants, there are many ways to infect a computer system, which can cause a lot of damage. The deep learning method is an effective countermeasure against various types and variants of malware.

With the spread of COVID-19 increasing telecommuting and internet use worldwide, traffic is exploding and cyberattacks are on the rise. Malware traffic also made up a significant portion of this traffic. As malicious codes become more intelligent and the methods of dissemination of malicious codes rapidly evolve, it is becoming difficult to prevent malicious codes.

Reviewing deep learning methods to classify and detect malware through the results of this study will help you find efficient deep learning methods for classifying and detecting malware on a host, network or smartphone.

ACKNOWLEDGEMENT

This work was supported by the research fund from Pyeongtaek University in 2020.

REFERENCES

[1] Malware, *https://en.wikipedia.org/wiki/Malware.*

[2] Malware Detection*, https://link.springer.com/referenceworkentry/10.1007%2F978-1-4419-5906-5_838.*

[3] Malware Classification, *https://paperswithcode.com/task/malware-classification.*

[4] Deep learning, *https://en.wikipedia.org/wiki/Deep_learning.*

[5] Autoencoder, *https://en.wikipedia.org/wiki/Autoencoder.*

[6] CNN*, https://en.wikipedia.org/wiki/Convolutional_neural_network.*

[7] RNN*, https://en.wikipedia.org/wiki/Recurrent_neural_network.*

[8] RBM*, https://en.wikipedia.org/wiki/Restricted_Boltzmann_machine.*

[9] Xin Wang, et. al, “A multi-task learning model for malware classification with useful file access pattern from API call sequence”, *https://arxiv.org/abs/1610.05945*.

[10] Omid E. David and Nathan S. Netanyahu, “Deepsign: Deep learning for automatic malware signature generation and classification,” In Proceedings of the 2015 International Joint Conference Neural Networks(IJCNN), Killarney, Ireland, pp. 1–8, 12–17 July 2015.

[11] William Hardy, et. al, “DL4MD: A deep learning framework for intelligent malware detection,” *In Proceedings of the International Conference Data Mining (ICDM)*, Barcelona, Spain, p. 61, 12–15 December 2016.

[12] Niall McLaughlin, et. al, “Deep android malware detection,” In Proceedings of the 7th ACM on Conference on Data and Application Security and Privacy, Scottsdale, AZ, USA, pp. 301–308, 22–24 March 2017.

[13] Bojan Kolosnjaji, et. al, “Deep learning for classification of malware system call sequences,” In Proceedings of the Australasian Joint Conf. on Artificial Intelligence, Hobart, Australia, pp. 137–149, 5–8 December 2016.

[14] Toshiki Shibahara, “Efficient dynamic malware analysis based on network behavior using deep learning,” In Proceedings of the 2016 IEEE Global Communications Conference(GLOBECOM), Washington, DC, USA, pp. 1–7, 4–8 December 2016.

[15] Zhenlong Yuan, et. al, ”Droid-sec:Deep learning in android malware detection,” *ACM SIGCOMM Computer Communication Review,* Vol. 44, Issue 4, pp 371–372, October 2014.