Trend of Malware Detection Using Deep Learning

Yoon-seon Lee
Hannam University
Daejeon, Republic of Korea
70, Hannam-ro, Daedeok-gu
+82-426297657
yoon7829@gmail.com

Jae-ung Lee
Hannam University
Daejeon, Republic of Korea
70, Hannam-ro, Daedeok-gu
+82-426297657
leejaeung1990@gmail.com

Woo-young Soh
Hannam University
Daejeon, Republic of Korea
70, Hannam-ro, Daedeok-gu
+82-426297657
wsoh@hnu.kr

ABSTRACT
According to recent security trends, there are more variations of existing malware than new type malware. These malwares are causing a lot of damage, such as encrypting users ' data to leak personal information, delete data, and make financial demands. Although many studies are being conducted to analyze malwares in order to respond to the rapidly increasing malware with such malicious purpose, current malware analysis methods are for obfuscation, virtual environment bypass, etc. To overcome these difficulties, Deep Learning method that analyzes and utilizes harmful codes is receiving spotlight recently. Therefore, this thesis introduces trends on how to analyze malware based on Deep Learning.

CCS Concepts
•Security and privacy → Intrusion/anomaly detection and malware mitigation → Malware and its mitigation

Keywords
Malware; Malware Detection; Deep Learning

1. INTRODUCTION
A malware is any form of software that is designed to harm a user with malicious purposes, such as leaking data from a computer or deleting files, regardless of his or her intention. The number of malwares with malicious purposes has increased significantly in recent years. In 2017, a type of Ransom Ware infected more than 300,000 computers worldwide, spreading fears of malware. According to Kaspersky Lab's report, the number of new Ransom Ware in 17 years versus 16 has decreased, but the number of strains has increased sharply in the last 10 years[1]. In particular, Figure 1 shows that the amount of malware found in recent years has increased dramatically, but that the number of new types of malware has been constant or decreasing.

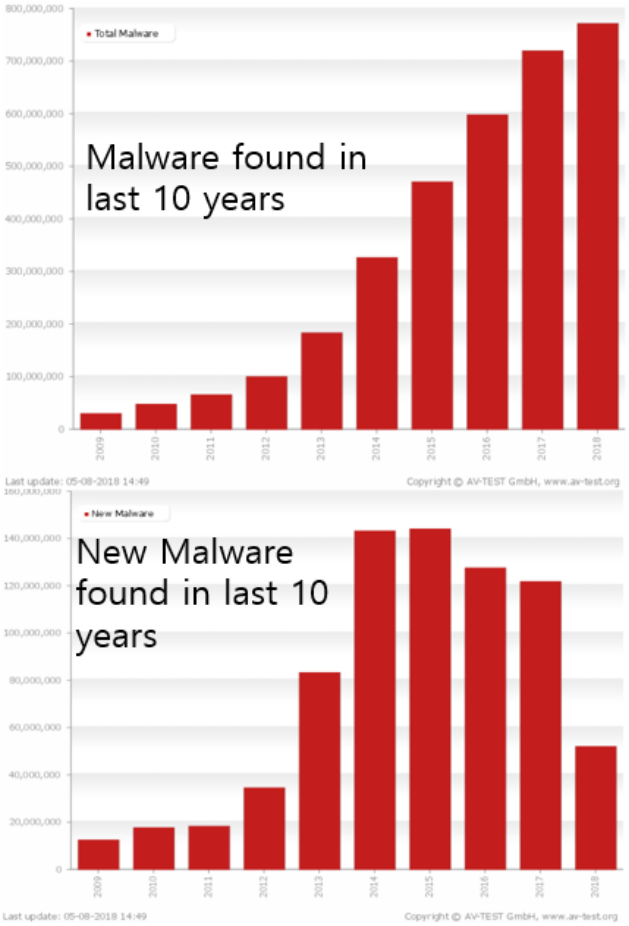

Figure 1. Malware Trend in Recent 10 Years

2. RELATED RESEARCH
2.1 Method for Malware Analysis
Depending on method, malware analysis can be divided into static and dynamic analysis. Although malware is largely divided into two methods depending on how it is analyzed, recent malware analysts are using hybrid analysis using both methods. As shown in Figure 1, hybrid analysis maximizes the benefits of dynamic and static analyses and reduces the disadvantages of each method of analysis. Static analysis uses debugger or reverse engineering techniques to analyze source code and the source code of the executable file to identify the function of the program and analyze the structure of the malware. The risk of infection is small because detailed analysis is possible compared to dynamic analysis and the

ICEMT 2018, July 2–4, 2018, Okinawa, Japan

ACM ISBN 978-1-4503-6525-3/18/07…$15.00
https://doi.org/10.1145/3206129.3239430

program is not executed directly[2]. However, the analysis may be difficult to automate and the analysis target is a malware. Dynamic analysis monitors behavior that is expressed through execution, and thus is not affected by the anti-buggers, such as obfuscation and packing. Although there is a high probability of detection of new malware and the advantage of reducing analysis time, malware that is produced only under certain conditions, such as the Jerusalem virus, is difficult to detect[2]. Therefore, many malware analysts have recently implemented dynamic analytics after deploying virtual analytics environments based on virtual machines.

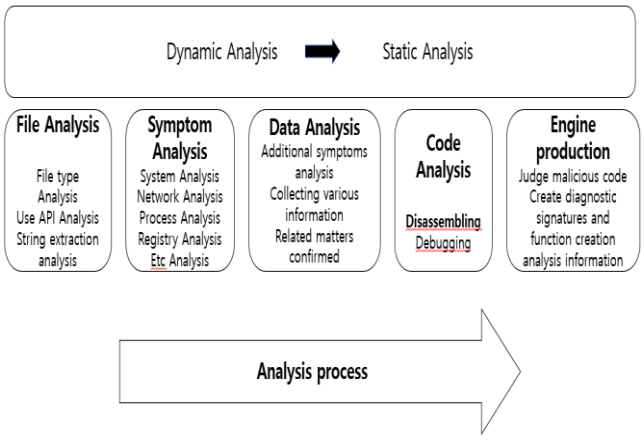

Figure 2. Malware Analysis Procedure

2.2 Deep Learning

Deep Learning is called deep learning as it is a kind of Machine Learning that imitates the human brain. Deep Learning, which means teaching a computer how to think, has moved away from research topics due to the problem of taking too much time to learn data. However, since AlexNet, who used CNN, a type of Deep Learning, won the 2012 image recognition algorithm competition, with 16.4 percent error rate, Deep Learning has drawn keen attention again. The structure of Deep Learning is shown in Figure 1. Input vectors are received as input from the input layer and given as silver layers. After learning in several layers of silver, you get results from the output layer. Deep Learning is able to produce accurate values as there are more data to learn through development of GPU that enables large data and parallel processing. Research continues to be done on Deep Learning, which allows computers to learn and produce results when certain data are available.

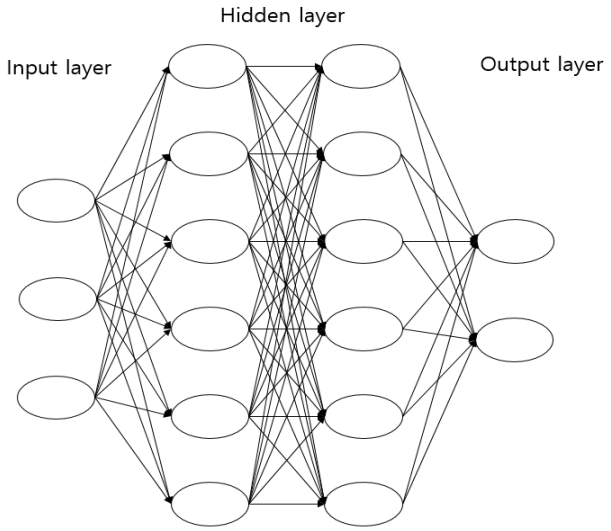

Figure 3. Deep Learning Structure

2.3 Big Data

Big Data is one of the hottest keywords in our society these days. Big Data, which is important in all sectors of society, is having a huge impact on human life. Big Data is a vast amount of data that goes beyond the traditional data unit and beyond the amount stored and managed by the typical database SW. Big data is not just a large amount of data, but data that can extract value from it.[3]

2.4 Fully Connected Layer

Fully connected layer is the layer that is associated with all the layers that close. As shown in Figure 5, the Fully connected layer consists of a combination of several Affine and ReLU layers, a final Affine layer, and a Softmax layer, and outputs the final result at the last softmax layer. In a fully connected layer, it is connected to neurons in all layers nearby. The number of outputs can be arbitrarily set. However, it has a problem that the shape of the input data can not be maintained[4].

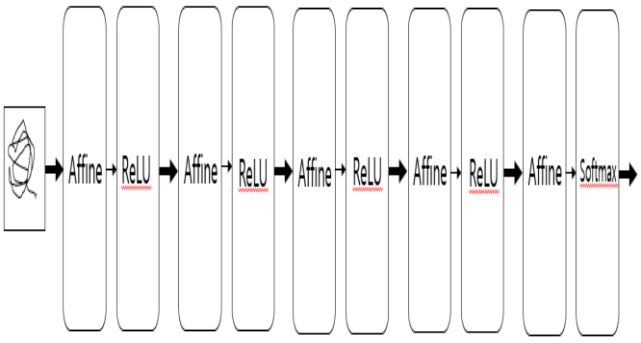

Figure 4. Example of a Fully-Connected Network

2.5 ReLU Function

The activation of the ReLU function, which emerged at the end of many attempts to improve the performance of Deep Learning, has recently been in the spotlight for surpassing the Sigmoid function. The formula of the ReLU used as an activation function is as follows. A very simple function, the ReLU function, is a function that outputs the input value as it is above zero, and the input value below zero. Figure a is a graph of the ReLu

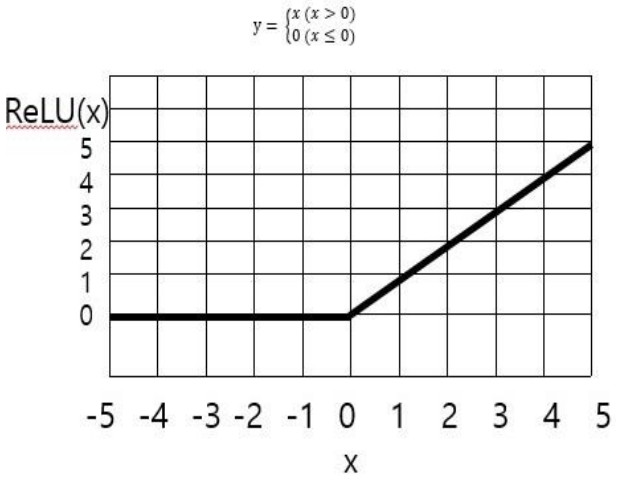

Figure 5. Graph of ReLU

2.6 Sigmoid Function

The sigmoid function is a commonly used activation function in neural networks. The sigmoid function serves as a converter that converts the transmitted signal and returns the output when the input is given from the neural network. Figure 6 is a graph of Sigmoid. The sigmoid function is a function of the sigmoid function.

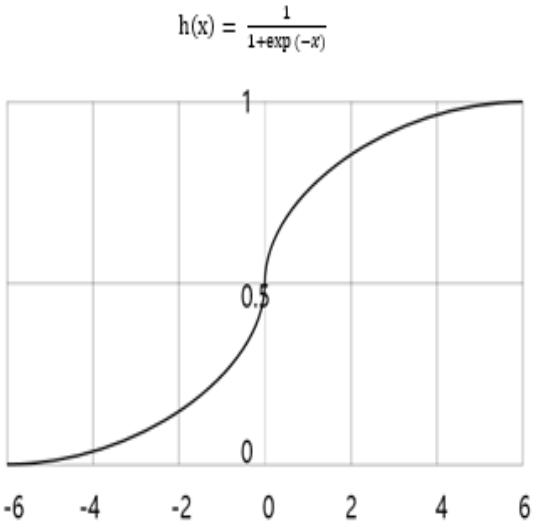

Figure 6. Graph of Sigmoid

2.7 CNN(Convolution Neural Networks)

CNN, a type of Deep Learning that is mainly used in image recognition and voice recognition, consists of a multi layer Perceptron. CNN can apply filters to input data to maintain the appearance of input data, a problem with Fully Connected Layer. By applying the values of input data and filters, CNN is typically divided into three classes by multiplying the corresponding elements into one another and then finding the sum. First is the Affine layer, which consists of a Fully-connected layer, to match input data and weights. Secondly, it consists of a synthetic product layer, which gives new values through the computation of the combined products of input data and filters, and a pooling layer that gives the average of the target area. The characteristics of CNN are extracted as shown in Figure 7 and 8. CNN, which receives input data and performs filter operations, has the advantage of being able to learn faster and with less error rate compared to previous techniques. As a result, a lot of research is going on[4].

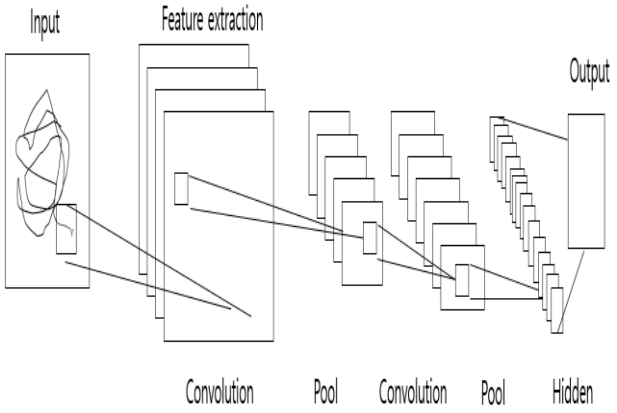

Figure 7. CNN Structure Example

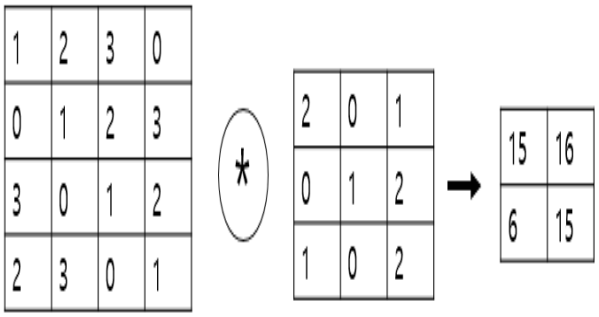

Figure 8. Example of Filter Operation

2.8 RNN(Recurrent Neural Networks)

It is a type of neural network algorithm consisting of RNN itself and a rotating structure. This cyclical structure allows RNN to change the resulting value due to the impact of the data previously processed when processing current data. It is usually used to process continuous data such as natural language or time series data. However, there are limitations in considering long-term dependency as a result of the occurrence of vanishing gradient problem in the data learning process. The structure of the RNN is shown in Figure 9.

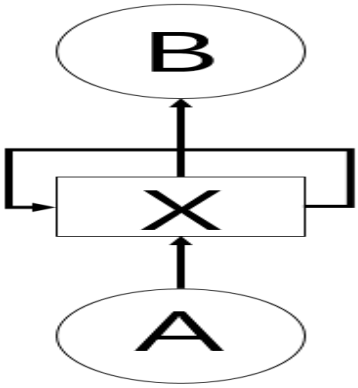

Figure 9. DNN Structure Example

3. METHODS FOR EXTRACTING MALWARE CHARACTERISTICS FOR DEEP LEARNING

Pretreatment work is essential for analyzing and classifying malware using Deep Learning. Several methods are used to use these pre-treatments.

First, do a dynamic analysis and extract the API (Application Programming Interface). API refers to the interface that the application provides to run, and there are methods to extract features such as files and data that use the API to execute malware. Another way to extract characteristics of malware is to image it. Figure 10 shows how to image. A variety of malwares that have changed parts of existing malware are characterized by images that give similar shape, and thus, malwares are analyzed and analyzed after analysis.

Finally, another way to extract the characteristics of malware is by using opcode extracted from the assembly conveyor of the file. Opcode defines the operations to be performed by the processor and identifies the data used by the command. Convert each opcode into a vector in one malicious file and convert it into a matrix of low sources to extract the feature[5]. Figure 11 shows a picture of CNN Embedding Layer using opcode.

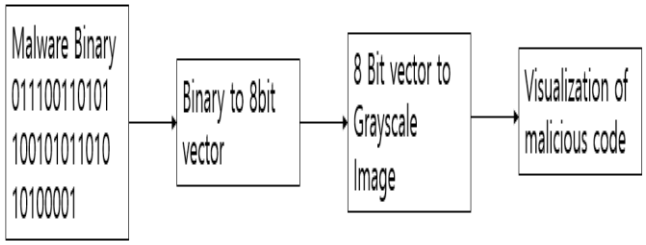

Figure 10. Process for Imaging Malware

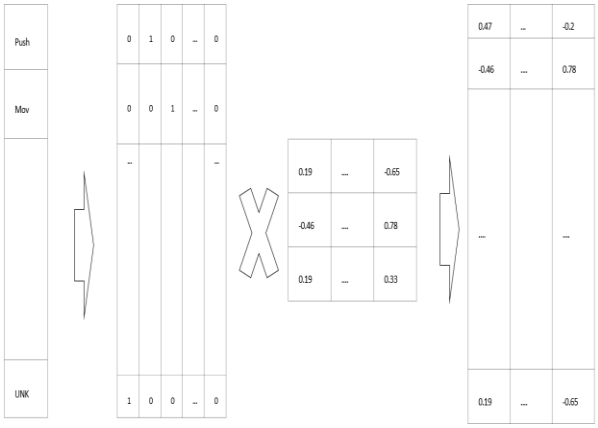

Figure 11. CNN Embedding Layer with Opcode

4. MALWARE DETECTION USING DEEP LEARNING

First of all, there is a method that uses API's characteristics. API refers to the interface that the application provides to run, which detects and learns the characteristics of the API used when the malware is executed by extracting the characteristics of the API.

Secondly, there is a way to determine if the file is malware or normal file, and if it is an electron, then family classification is possible. Most of the discovered malware is a variant of the existing malware, not a new type, and the malware, which has modified only part of the code, has similar images. These characteristics were used to extract and image binary code of malware and then CNN[6].

Third, malware is used using the operation code. Opcode is the part that specifies the computation action with the command code, which is the key part of the machine command. There is a method to extract characteristics from the opcode of malware and learn them using CNN techniques to classify malware[5].

5. DEEP LEARNING BASED DETECTION RESEARCH

Research that detects harmful codes based on Deep Learning is already under way.

For the first time, " Android Malware Analysis Technology Study based on Naïve Bayes " suggested a method to analyze malware based on machine learning algorithms called Naïve Bayes. A statistical approach to the API was used to determine normal and malicious applications and extract raw data from the method information area of the .dex file. The results tested on the Naïve Bayes model detect malware with 85 % accuracy[8].

Second, the " image-based artificial intelligence to protect big data from malware " study showed that when malware is analyzed, the variant malware has a similar image, although some of it may change from existing sources. The results of learning and classifying malware binary files through CNN showed 91.7 percent accuracy and detection of malware[9].

Third, the " Family classification through malware imaging based on Microsoft Transformation Neural Network " study also used an 8-bit grid image for family classification without executing malware. Microsoft datasets recorded 96.2 % accuracy for nine family classifications, and 82.9 % accuracy for 27 family classifications for VXHeavens datasets[10].

Fourth, in the " A Study on the Machine Learning Proposal for Mobile Malware Detection " study, we used the Decisional Tree and Entropy, Random Forest, Support Vector Machine (SVM), etc. 181 APIs and String information were selected as characteristics by extracting characteristics that show the characteristics of each app. In addition, API and String information were selected and extracted as characteristics. As a result, it detected harmful apps with detection rate at 83 % on average[10].

Finally, the study "Study on DNN Based Android Malware Detection Method for Mobile Environment" suggested that a static analysis of an application and a dynamic learning method are combined to directly check the application information within the smartphone[11].

All of the preceding studies can't be evaluated for accuracy using different datasets. Because Deep Learning can achieve better results if it learns with more data, it can produce different results if it has more data.

6. CONCLUSION

In this thesis, we introduced trends on detecting harmful codes with Deep Learning. In order to detect malwares by utilizing Deep Learning, data that will be used for Deep Learning models must first be extracted and data extracted must be developed and taught. Accordingly, malware analysts have proposed malware processing techniques to extract signatures, APIs, and opcodes of malware in various ways to learn data. It is also possible to automatically extract complex and difficult characteristics from

simple data characteristics of malwares compared to existing analysis and classification. Deep Learning's method of detection and classification of malwares by using Deep Learning requires Deep Learning models that can more efficiently and accurately classify and detect harmful codes so that they can respond to recent increase in number of harmful codes. Furthermore, the accuracy of the preceding studies may depend on the amount of data, depending on the dataset. Therefore, it proposes to unify datasets that will be used for malware analysis research using Deep Learning.

7. ACKNOWLEDGMENTS

This work was supported by 2018 Hannam University Research Fund

8. REFERENCES

[1] Kaspersky lab korea. *report 26 % of the fastest evolving security threat, ransomware, is targeted at companies* http://news.kaspersky.co.kr/news2017/12n/171201.htm

[2] Michael Sikorski, Andres Honig. 2012. *Practical Malware analysis*.

[3] Kyo-Il Chung, Han-na Park, Boo-Geum Jung, Jong-Soo Jang, Myung-Ae Chung. 2012. *Big Data and Information Security*. Korea Institute of Information Technology Magazine, 10(3), 17-22.

[4] *Deep Learning from Scratch*. 2016. O'Reilly Japan

[5] Daniel Gibert. *Convolutional Neural Networks for Malware Classification*. 2016. Master Thesis, Universitat de Barcelona

[6] Jae Woon Park. 2016. *Deep Learning Based Malware Detection Using API Features.* Department of IT Convergence Information Security Graduate School of KonKuk Universi http://konkuk.dcollection.net/public_resource/pdf/000002316768_20180522134444.pdf

[7] Jun-ho Hwang, Tae-jin Lee. 2017. *Android Malware Analysis Technology Research Based on Naive Bayes*. Journal of the Korea Institute of Information Security & Cryptology, 27(5), 1087-1097.

[8] Hae Jung Kim, Eun Jun Yoon. 2017. *Image-based Artificial Intelligence Deep Learning to Protect the Big Data from Malware*. Journal of the Institute of Electronics and Information Engineers, 54(2), 76-82."

[9] Seonhee Seok, Howon Kim. 2016. *Visualized Malware Classification Based-on Convolutional Neural Network.* Journal of the Korea Institute of Information Security & Cryptology, 26(1), 197-208.

[10] Sungtaek OH, Woong Go, Junhyung Park. 2017. *A Study on the Machine Learning Algorithm for Mobile Malware Detection*." Korea Information Science Society 1111-1113.

[11] Jin Hyun Yu. 2017. *Study on DNN Based Android Malware Detection Method for Mobile Environment.* Department of Information Security Graduate School of Information Security Korea University