Scientific Journal of Impact Factor (SJIF): 4.72

e-ISSN (O): 2348-4470
p-ISSN (P): 2348-6406

International Journal of Advance Engineering and Research Development

Volume 4, Issue 11, November -2017

A Study on High-Speed Malware Classification Using Clustering and Neural Network

Dae-hoon Yoo, Bo-min Choi, Kyung-han Kim, Hong-koo Kang, Jun-hyung Park

Korea Internet & Security Agency

Abstract — *Cyber-attacks are keep increasing, and most of these attacks begin with malicious code. Therefore, in order to reduce the damage caused by cyber-attack, malicious code should be able to be detected and analyzed quickly. According to the AV-test(2017), most of the malicious code found is a variant of existing malware. Therefore, if we can identify relationship between newly discovered malware and existing malware, the damage caused by cyber-infringement accidents. According to research, most of the malicious code found is a variant of existing malware. Therefore, if we quickly identify relationship between newly discovered malware and existing malware, then we can reduce the damage caused by cyber-attacks. In this paper, we propose new method to classify malwares at high-speed. First K-means algorithm is used to cluster similar malwares. Then, train a neural network using the clustering result. After training the network, we can classify malwares at high speed. As a result of experiment with 49,561 malwares that actually distributed, 84.18% of them are classified into 315 clusters, and the average similarity of clusters was 97.06%. And accuracy of classification is over 90%.*

Keywords- *Malware Classification, K-means Clustering, Deep Learning*

ACKNOWLEDGMENT

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No. 2016-0-00081, The Development of Integrated Malware Life-cycle Profiling and Attack-group Identification Technology)

I. INTRODUCTION

Cyber-attack is rapidly increasing, and in many cases it starts with malware. Since most of the malware is a variant of existing malware, if we can quickly identify relationship between newly discovered suspicious code and existing malware, we are able to respond to the newly discovered code within a reasonable time.

II. PROPOSED METHOD

We do not know whether files collected from a real environment are malicious or not. Even in case of suspicious file to be malicious, it is difficult to know the family to which the malicious code belongs until the diagnosis result is obtained by using a vaccine program. In this section, we propose a framework that can classify malwares which have similar behavior. The proposed method identifies and classifies malwares having similar behavior based on the similarity among collected malwares on the absence of any information.

2.1. Overview

The proposed method consists of 1) step of normalizing the behavior information, i.e. API call information, by malicious code, 2) step of generating a label by clustering based on the degree of similarity between pieces of behavior information and 3) step of learning the classification model using the labeled data.

2.2. Preprocessing the API call sequence using N-gram & Feature Hashing

API sequence called by malicious code is the most typical behavior information of the malicious code. Since the type and number of API calls differ by malicious code, it is difficult to use API call sequence as an input of clustering algorithm and neural network algorithm. Therefore, the proposed method normalizes the API call sequence of malicious code as follows.

First, an API call sequence having an arbitrary length is converted into a vector having a predetermined length using an n-gram algorithm. Next, the vector transformed by the n-gram algorithm is compressed by the feature hashing algorithm. There are hundreds of APIs that malicious code can call, but the kinds of API that a single program calls are limited. Therefore, a vector that converts an API call sequence of malicious code into n-gram is very long but sparse. Clustering or neural network based classification algorithms may not be efficient when using vectors with these characteristics as input. Therefore, by compressing the existing vector by Feature Hashing, the information in the existing vector is saved as much as possible while reducing the length and density.

2.3. Grouping the Similar Behavior Malware using Repeated K-means Clustering

One of the ways to group similar data properly when there is multiple data is the clustering algorithm, and the K-means algorithm is the most representative clustering algorithm. The K-means algorithm has the advantage of processing a large amount of data at a high speed, but it has disadvantages such as a large difference between good and poor classification results, and sensitivity to outliers. One of the difficulties in applying the K-means algorithm is that it does not know how to determine the optimal K for any input. In the proposed method, the K-means algorithm is applied repeatedly to group similar malicious codes with the proper number.

First, considering the number of malicious codes to be clustered, K is determined and K-means clustering is performed, and the similarity of each malicious code with its own cluster center is measured. Next, the cluster size (the number of malicious codes) and the average cluster similarity (average of malicious code and center similarity) are calculated. All clusters belong to one of the following three cases. Filtering or iterative clustering is performed for each case.

case 1) The size is larger than the size threshold and the similarity average is higher than the similarity threshold.

i. The malicious codes whose similarity to the cluster center is lower than the threshold are excluded from the cluster.
ii. Recalculate the cluster center against malware remaining in the cluster.
iii. Compute the similarity of the newly calculated cluster centers and the remaining malicious codes in the cluster.
iv. Repeat steps i to iv until the center similarity of all malicious codes is above the similarity threshold.

case 2) The size is larger than the size threshold and the similarity average is lower than the similarity threshold.

i. Considering the size of the current cluster, the appropriate K is determined, and the K-means clustering is performed only on the current cluster.
ii. For all malicious codes, we measure the similarity with the cluster center to which they belong, and obtain the cluster size and similarity average.
iii. The clusters are classified according to the size threshold and the similarity threshold, and filtering (i.e. case 1) is performed for clusters having a size larger than the size threshold and a similarity larger than the similarity threshold. Clustering (i.e. case 2) is performed for a cluster whose size is larger than the size threshold and the degree of similarity is smaller than the similarity threshold.

case 3) The size is smaller than the threshold.

i. Classify to the new variants of malware.

2.4. Neural Network Structure for New Variants of Malware Detection

When classifying a malicious code variant group without any prior knowledge, it is difficult to know that how many groups of variants are exists, and there is always the difficulty that new variant malicious codes can appear. In these days, one of the hot methods in classification is deep learning. Deep learning is a neural network algorithm, in which multiple neurons are placed in layers and connected by weights to represent the relationship between input and output. In the proposed method, the labeling information identified by the iterative K-means clustering of 3.3 is using to train the deep learning network to classify the malicious code at high speed. Since the Deep learning network has a limit to classify only a predetermined number of classes, the proposed method has extra nodes in the Output layer and activates the extra node whenever a new malicious code variant group appears. Although there are many ways to find a new malicious code variant group, the proposed method uses a repetitive clustering method of 3.3 for malicious codes not included in the existing group.

III. EXPERIMENTAL RESULT

The following is a high-speed classification of 49,561 malicious codes using the proposed method.

3.1. Clustering Result

Table 1. Clustering Result

		Smilarity											sum
		0.80	0.82	0.84	0.86	0.88	0.90	0.92	0.94	0.96	0.98	1.00	
Size	10			1		1	2	6	9	6	12	51	88
	20					1	3	4	4	7	9	34	62
	30				1		1	2	1	3	6	15	29
	40						1	1	2	4	3	6	17
	50								2	3	3	4	12
	60							2	1	1	1	4	9
	70						1		1	1		3	6
	80									1	1	5	7
	90								1		2	3	6
	100											3	3
	200						1	2	1	3	2	16	25
	300								1	1	3	6	11
	400							1	1		2	6	10
	500										2	3	5
	600									2	1	4	7
	700											1	1
	800							1				2	3
	900										1		1
	1,000										1	2	3
	1,500							2			2	3	7
	2,000								1		1		2
	2,500												
	3,000										1		1
sum				1	1	2	9	21	25	32	53	171	315

3.2. Classification Result

After 100 epoch, Training Accuracy is 95.50%, and it takes 5,984.73 seconds.

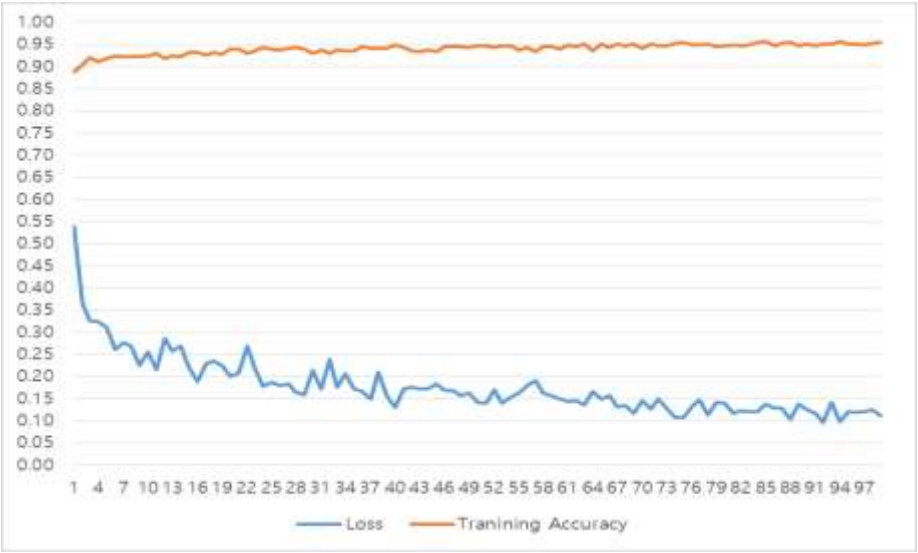

Figure 1. raining Accuracy

IV. CONCLUSION

In this paper, we proposed a method to classify malicious codes at high speed using K-means clustering and deep learning. It was possible to classify a large number of malicious codes in a short period of time without any prior knowledge, and it was also possible to automatically detect new malicious codes. In the future, methods of preprocessing malicious code 's behavior information, labeling method, and classification network improvement should be studied.

REFERENCES

[1] AV-Test, Mlware Statistics & Trends Report, 2017, <www.av-test.org/en/statistics/malware>
[2] OH, Sungtaek; GO, Woong; LEE, Taejin. A Study on The behavior-based Malware Detection Signature. In: International Conference on Broadband and Wireless Computing, Communication and Applications. Springer International Publishing, 2016. p. 663-670.
[3] CHO, In Kyeom, et al. Malware Similarity Analysis using API Sequence Alignments. J. Internet Serv. Inf. Secur., 2014, 4.4: 103-114.
[4] LEE, Taejin, et al. Automatic malware mutant detection and group classification based on the n-gram and clustering coefficient. The Journal of Supercomputing, 2015, 1-15.

