EasyChair Preprint

№ 3177

A Hybrid Deep Learning Approach for Detecting Zero-Day Malware Attacks.

Shaik Moin Sharukh

April 16, 2020

A hybrid deep learning approach for detecting zero - day malware attacks.

Shaik Moin Sharukh [1]

[1] VNR Vignana Jyothi Institute of engineering and technology, Hyderabad, India
moinsharukh001@gmail.com

Abstract. Begun in 1988, malware detection continues to be a challenging research topic in this epoch of technology. The exponential rise of IoT devices and its consumers has parallelly increased the number of security breaches in recent times, posing a major security concern. Research studies in malware detection analysis have proved that both dynamic and static analyses are time-consuming, inefficient and ineffective to detect novel malware signatures. The cybercriminals make use of evasive techniques like polymorphism and code obfuscation to alter the malware behavior rapidly and bypass malware detection. To countermeasure the cyber-attacks, machine learning algorithms (MLA's) have come into the picture. The feature learning technique used by MLA's to detect novel malware signatures turns out to be time-consuming. To bypass the feature engineering phase, we introduce the deep learning methodologies such as long short-term memory (LSTM) and convolutional neural networks (CNN). We made use of the binary malware datasets to train the algorithms, and once the malwares are detected they are classified and categorized into their respective malware families by means of deep image processing techniques. The results obtained in this paper showcases the Brightside of the deep learning architectures by outperforming the machine learning algorithms.

Keywords: Malware detection, deep learning, machine learning, cybercrime, image processing.

1 Introduction

The twentieth century has witnessed a sheer dominance of the information society posing a major security concern. With the revolution lead by the information society, the production of IoT devices got amplified along with the increasing number of users. These billions of IoT devices generate an enormous amount of data, paving a path to data breaches. Since the mainstream users stand unaware of the default security settings in their devices the cybercriminals utilize the vulnerabilities to attack the devices with various malware and steal confidential data to obtain financial gains [1]. Malware is simply a code engendered by cybercriminals to launch cyber-attacks and gain unauthorized access to various devices in a network. It has numerous variants like trojan, worm, ransomware, command and control bot, adware, virus and spyware [2]. Malware

detection remains an unremitting process until the malware authors stop developing novel evasion techniques.

1.1 Research Background

The inception of anti-virus software happened in 1987, to detect the existence of the first malware. Signature-based detection remained to be the foremost technique used in the anti-virus software to understand the behavior of the malware files. The signature-based detection techniques evidenced to be ineffective to detect novel malware signatures, as they failed to bypass malware evasion techniques like stegosploit, code obfuscation, and code encryption [3]. To reverse engineer the novel malware signatures, the signature-based detection technique requires deep domain-level knowledge which is time-consuming.

To countermeasure the malware evasion techniques, security researchers introduced machine learning algorithms to detect and categorize malware into their respective families. The machine learning algorithms (MLA's) employ the domain level engineering and feature selection approaches to generate a separating plane between malware files and benign files.[4]-[5]. The features employed by MLA's are attained from dynamic and static analysis. If a code is inspected during execution, it falls into the class of dynamic analysis and if the same code is examined without execution it falls into the class of static analysis. When compared, dynamic analysis outperformed static analysis in distinguishing benign and malware signatures but has issues in time complexity.

For a machine learning algorithm to be successful, it requires rigorous training under various patterns of malware. Moreover, MLA's evidenced fading of outputs when enormous data is dumped whereas deep learning seizures novel patterns and generates a relation with the longstanding patterns to attain better performance and results [6]. The lack of efficiency and accuracy in MLAS's inspired the current research paper to explore the deep learning algorithms and propose an efficient architecture for malware detection.

1.2 Deep Learning Architectures

Artificial intelligence acts as the fountainhead for deep learning and machine learning architectures, as it has analogous functionalities that of a human brain. In this epoch of technology, the IoT devices generate huge amount of data and machine learning algorithms require domain level knowledge to preprocess the data and track down the malware. The deep learning architectures such as recurrent neural network (RNN) and convolution neural network, possess the competence to understand and process data in large amounts unlike machine learning algorithms [6].

In this paper, we are using deep learning algorithms such as CNN for geospatial data and long short-term memory (LSTM) to detect, classify and categorize malwares into their respective malware families.

2 Implementation Methodology

In this paper, we implement several deep learning methodologies such as deep static analysis, deep image processing technique and proposed architecture for detecting malicious malware binaries.

2.1 Detection of malware binaries using deep learning methodology

The efficiency of specific machine learning algorithms such as Support vector machine (SVM) algorithm[13], Random forest, Decision Tree, Naive Bayes (NB), Logistic Regression, K-Nearest Neighbours (KNN) is evaluated along with two deep learning algorithms. The below figure 1 depicts the working mechanism of the deep learning architecture used for splitting benign and malware binaries.[9]

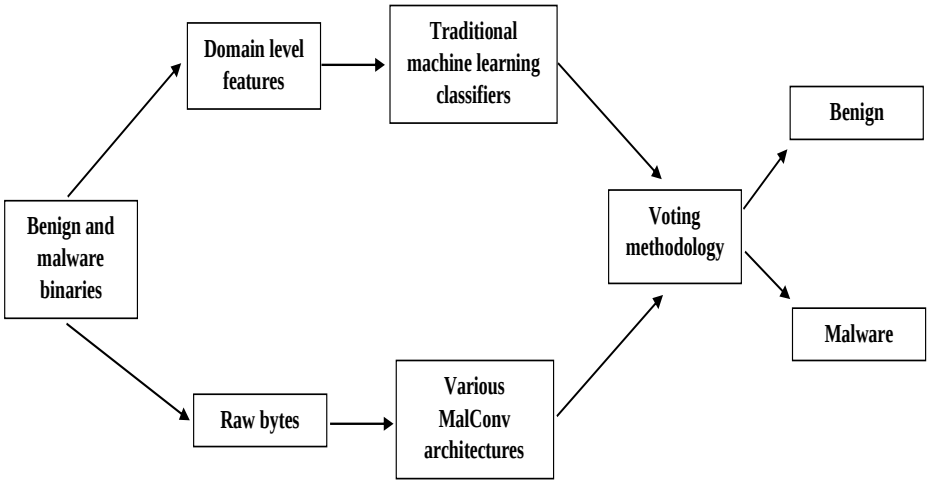

Fig. 1. Deep learning methodology for splitting benevolent and malicious malware binaries.

Initially, the dataset comprising both benign and malware binaries is uploaded where both raw bytes and domain level features are extracted from the dataset. The domain level features such as file name, file size, hashes, MD5 checksum, etc are forwarded to the traditional MLA's for malware classification and detection. In a parallel method, the raw bytes are subjected to the deep learning methodologies for detecting, classifying and categorizing malware binaries [5]. The voting methodology adopts both algorithms and finally classifies the binaries into two classes namely benign and malware. The detected malwares are categorized into respective malware families using deep image processing technique dependent on deep learning methodology.

2.2 Categorization of detected malware using Deep image processing technique dependent on deep learning methodology

The CNN alongside LSTM [16] form a hybrid pipeline to categorize malwares based on image processing technique [7]. In this paper we use visualization for categorizing malware into their respective families by evading the feature engineering segment. Unlike static analysis, the image processing technique uses raw bytes of information which makes it faster and added to that it can completely evade the execution phase [8] and [9]. The proposed image processing technique is compatible with malwares derived from various operating systems like windows, Linux, Android, etc. The below figure 2 portray the deep image processing technique.[9]

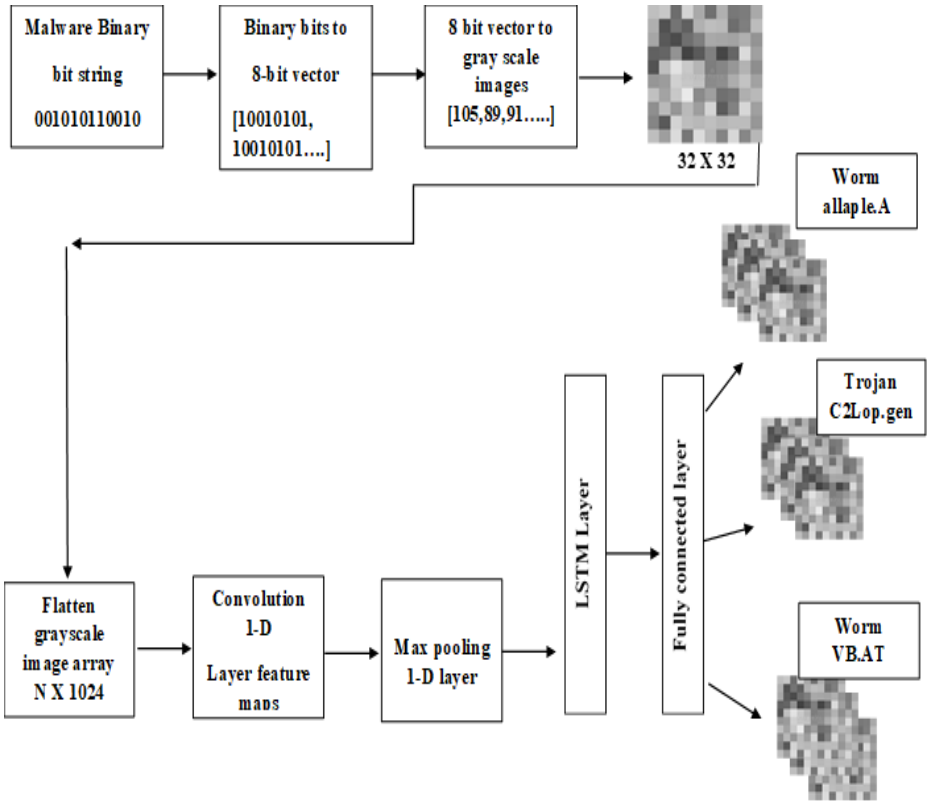

Fig. 2. Deep learning architecture based on image processing.

The malware binaries in the dataset exhibit three colors namely black, white and gray [17]. The black, white colors in the grayscale image portray 0 and 255 respectively. Malware binaries with transitional shades of gray fall in between [0-255]. If the image comprises only black color it means that the grayscale image holds only 0's in it and if the image is subjected to white color it portrays that the majority part of the image comprises the number 255 [7].

Description of Dataset.

In this paper, we used maling dataset which comprises 9,339 malware signatures categorized in 25 different families. For training and testing the dataset we have separated the dataset into two sections.[11] The primary segment comprises 80% of malware data, used for training the dataset and the subsequent segment comprises the remaining 20% of the malware data for testing the malware binaries [9]. The two parts of the dataset comprise malwares signatures. Initially the data was existing in the format of malware binaries later these were converted into matrix format i.e. 8-bit unsigned integer. As discussed above, these matrices are visualized as a picture or gray-scale image. After visualizing the picture in a 2D matrix it is transformed into a 1D vector form which results in the formation of a 1024 sized array [9].

3 Proposed Architecture – DLMDN

An outline of our proposed architecture DeepLearningMalwareDetectionNetwork (DLMDN) is depicted in Figure 3. The proposed framework has a sequential procedure partitioned into five steps, to detect the malwares [9],[10].

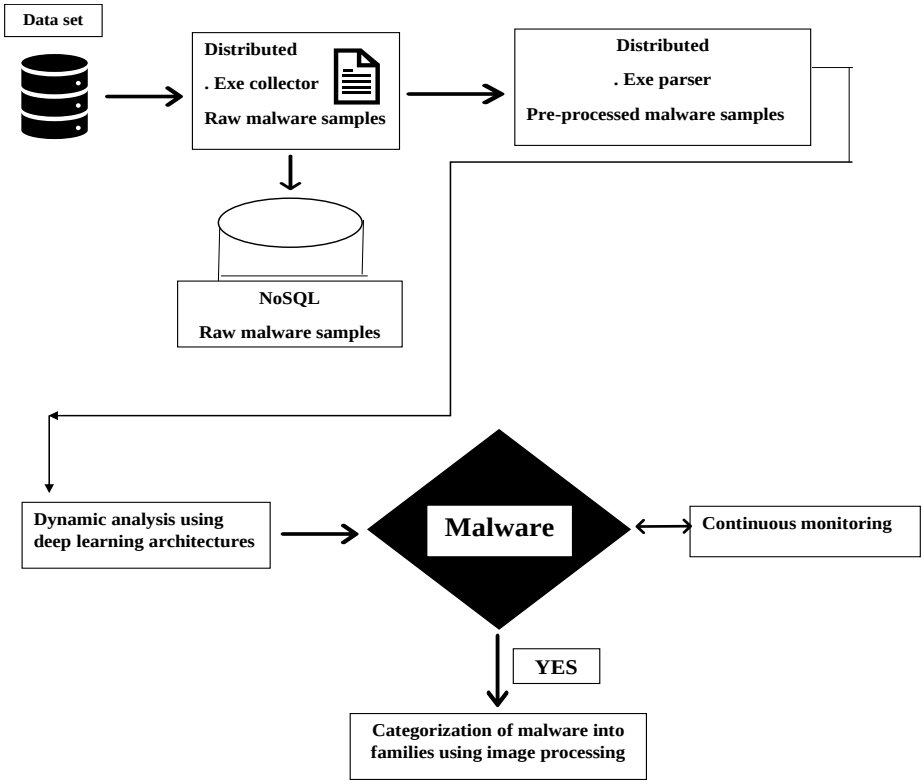

Fig. 3. Proposed architecture-DLMDN

Initially the collected data from the malimg dataset is partitioned using. Exe parser [3] and in the subsequent step the malware samples are subjected to pre-processing. To distinguish between the benevolent and malicious malware binaries the pre-processed data is progressed to a voting methodology based on machine learning and deep learning algorithms.[12]

As shown in the above figure, the detected malware is continuously monitored. Using the deep image processing technique, the detected malwares obtained in the binary format are transformed into matrix format i.e. 8-bit unsigned integer [14]. These unsigned 8-bit integers are visualized as greyscale images in a 2D matrix which is converted into 1-D vector format resulting in a 1024 sized array. Upon thorough training, these malwares are categorized into their respective malware families [15].

4 Results

The results obtained in this paper showcases the Brightside of the deep learning architectures by outperforming the machine learning algorithms. As shown in the below figure 4, the convolutional neural network (CNN) algorithm has surpassed many MLA's algorithms such as Support Vector Machine (SVM), Naive Bayes, Decision-tree, etc.

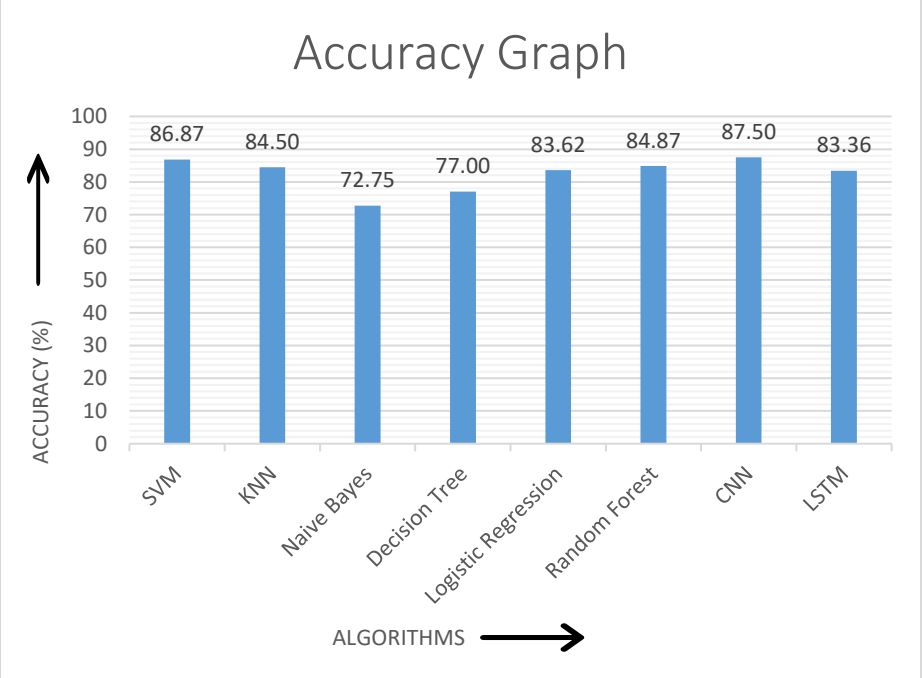

Fig. 4. Accuracy graph of MLA and Deep learning algorithms.

The convolutional neural network (CNN) has gained the uppermost accuracy rate amongst all algorithms in detecting malware samples. The superlative algorithm for detecting novel malware signatures is determined by assessing major factors like accuracy rate of detection, precision, recalling factor and F-score. Below table 1 depicts the obtained results for malware detection.

Table 1. Detailed Test results

Algorithms	Accuracy (%)	Precision (%)	Recall (%)	F-Score (%)
SVM	86.87	86.77	86.80	86.79
KNN	84.50	84.49	84.55	84.52
Naive Bayes	72.75	72.70	72.79	72.77
Decision Tree	77.00	77.10	76.35	77.26
Logistic Regression	83.62	83.65	83.70	83.68
Random Forest	84.87	84.90	84.87	84.86
CNN	87.50	87.49	87.51	87.50
LSTM	83.36	83.38	83.35	83.34

5 Conclusion

We conclude the paper by proposing a novel framework called DLMDN for evaluating both the MLA's and deep learning methodologies. The proposed framework has a sequential procedure partitioned into five steps, to detect the malwares. It consists of collecting raw malware samples, parsing the data, pre-processing, detecting and categorizing the malware into respective malware families. The categorization of malwares is performed by image processing technique. This paper has proved the supremacy of deep learning methodologies over MLA's in terms of accuracy for detection of novel malwares, precision rate, recalling factor and F-score.

6 References

1. M.Tang, M. Alazab, and Y. Luo, “Big data for cybersecurity: Vulnerability disclosure trends and dependencies,”' IEEE Trans.Big Data, to be published.
2. M. Alazab, S.Venkatraman, P.Watters, M. Alazab, and A. Alazab, “Cybercrime: The case of obfuscated malware,” in Global Security, Safety and Sustainability & e-Democracy (Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering), vol. 99, C. K. Georgiadis, H. Jahankhani, E. Pimenidis, R. Bashroush, and A. Al-Nemrat, Eds. Berlin, Germany: Springer, 2012..
3. E. Raff, J. Barker, J. Sylvester, R. Brandon, B. Catanzaro, and C. Nicholas. (2017). Malware detection by eating a whole .exe” [online] available https://arxiv.org/abs/1710.09435.” e
4. M. Rhode, P. Burnap, and K. Jones, “Early-stage malware prediction using recurrent neural networks,” Comput. Secur., vol. 77, pp. 578594, Aug. 2018
5. T. Shibahara, T. Yagi, M. Akiyama, D. Chiba, and T. Yada, “Efficient dynamic malware analysis based on network behavior using deep learning,” in Proc IEEE GlobalCommun.Conf (GLOBECOM), Dec.2016, pp. 1-7.
6. Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, no. 7553, pp. 436_444, 2015.
7. S. Ni, Q. Qian, and R. Zhang, “Malware identification using visualization images and deep learning,” Comput.Secur., vol. 77, pp. 871_885, Aug. 2018.
8. G. Sun and Q. Qian, “Deep learning and visualization for identifying malware families,” IEEE Trans. Dependable Secure Comput. to be published.
9. R. Vinaykumar, M. Alazab, K. P. Soman, P. Poornachandran, and S. Venkatraman, “Robust intelligent malware detection using deep learning”, IEEE Access. vol. 7, April. 2019
10. E. Raff, J. Sylvester, and C. Nicholas, “Learning the PE header, malware detection with minimal domain knowledge," in Proc.10th ACM Workshop Artif.Intell.Secur. New York, NY, USA: ACM, Nov. 2017, pp. 121_132.
11. G. Conti, E. Dean, M. Sinda, and B. Sangster, “Visual reverse engineering of binary and data files,” in Visualization for computer. Berlin, Germany: Springer, 2008, pp. 1–17.
12. M. Alazab, ‘‘Profiling and classifying the behavior of malicious codes,” J. Syst. Softw., vol. 100, pp. 91–102, Feb. 2015.
13. S. Huda, J. Abawajy, M. Alazab, M. Abdollalihian, R. Islam, and J. Yearwood, ‘‘Hybrids of support vector machine wrapper and filter based framework for malware detection,’’ Fu ture Gener. Comput. Syst., vol. 55, pp. 376–390, Feb. 2016.
14. M. Krcál, O. Švec, M. Bálek, and O. Jašek. (2018). Deep Convolutional Malware Classifiers Can Learn from Raw Executables and Labels Only. [Online]. Available: https://openreview.net/forum?id=HkHrmM1PM
15. J. Saxe and K. Berlin, ‘‘Deep neural network based malware detection using two dimensional binary program features,’’ in Proc.10th Int.Conf.maliciousUnwanted Softw.(Malware), Oct. 2015, pp. 11–20.
16. R. Pascanu, J. W. Stokes, H. Sanossian, M. Marinescu, and A. Thomas, ‘‘Malware classification with recurrent networks,’’ in Proc.IEEE Int.Conf.Acoust.,Speech Signal Process.(ICASSP), Apr. 2015, pp. 1916–1920.
17. L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, ‘‘Malware images: Visualization and automatic classification,’’ in *Proc. 8th Int. Symp. Vis. Cyber Secur.* New York, NY, USA: ACM, Jul. 2011, p. 4.