Analysis of Android Malware Detection Techniques in Deep Learning

Neetu Agrawal[1], Vipin Jain[2], Raju K Ranjan[3]
Department ofComputer Science &Engineering, Swami Keshvanand Institute of Technology, Management & Gramothan Jaipur-302017 (INDIA)
Department ofInformation Technology, Swami Keshvanand Institute of Technology, Management & Gramothan Jaipur-302017 (INDIA)
Department ofComputer Science &Engineering, Delhi Technological University, New Delhi-110042 (INDIA)
Email -neetu162@gmail.com[1], vipin@skit.ac.in[2]
Received 27.08.2020 received in revised form 15.10.2020, accepted 17.10.2020

Abstract: Over the years, developments in smart phone technology has boost up their use among users. This has captivated malware authors' attention. Malware attacks in various forms has troubled users by stealing their personal information, banking information and much more. Android users have been strained most because of Android's open nature. Throughout this time, efforts have been made to devise software and methods to detect android malwares. Starting from anti-virus software to Machine Learning and now Deep Learning, researchers have put forward various techniques to get to grips with the problem. Many Deep Learning Techniques have been put forward like Deep Neural Network, Convolutional Neural Network, Recurrent Neural Network, Deep Belief Network and Autoencoders. This paper looks at and analyzes supervised Deep Learning classifiers to detect Android malware.

Keywords– Deep Learning, Android Malware, CNN, Neural Network.

1. INTRODUCTION

With the world-wide revolution in smart phones, most of the important tasks were happening on phones including financial transactions, social engineering, and personal data. This openly called for malware attacks. Because of the open nature of Android OS, it was more prone to malware attack. Innumerable techniques and software have been developed till now that includes anti-virus software, behavior-based models, machine learning and deep learning techniques.

Antivirus software use signature-based methods. It involves generating signatures of existing malware and match those signatures with the given application to identify it as malware or benign. But these methods are too feeble to face code obfuscation or repackaging. Code obfuscation changes malware signatures which enables them to go undetectable.

To address this problem, behavior-based machine learning methods came to the rescue. Machine learning methods are intelligent and automatic methods that can detect unseen malware after being trained. Machine learning techniques build classifiers that perceive malware from among benign applications.

To further improve the efficiency of Machine Learning techniques, Deep Learning techniques were introduced which are a branch of Machine Learning. Many researchers have developed frameworks using different machine learning and deep learning algorithms like Decision Trees, Naïve Bayes, Support Vector Machine (SVM), Deep Neural Network (DNN), Convolutional Neural Network (CNN) and Autoencoders. The paper provides a brief outline of all the current algorithms which have been used by researchers for developing frameworks to detect android malware focusing on deep learning techniques.

Rest of the paper is organized as follows. Section 2 gives information about Android malware and its variations. Section 3 explains about Deep learning and its categorizations: static, dynamic and hybrid analysis. Section 4 gives literature survey of various papers that have developed methods and frameworks for android malware detection using Deep learning. Section 5 concludes the paper.

2. ANDROID MALWARE

Malware is a malicious program written with the intention of causing harm to our devices and personal information. Android is attacked most because of its open source OS and its popularity. According to a global OS market share for the year 2020 [2], market share of android is hover around 85%. Distribution of market share of various mobile OS is shown in figure 1.

With the increase in Android's market share, types of malware attacks have increased and along with that, new methods of malware detection are

also proposed by experts every now and then. Types of mobile malwares that can harm the mobile devices are spyware, ransomware, virus, trojans, worms, bot processes and crypto mining.

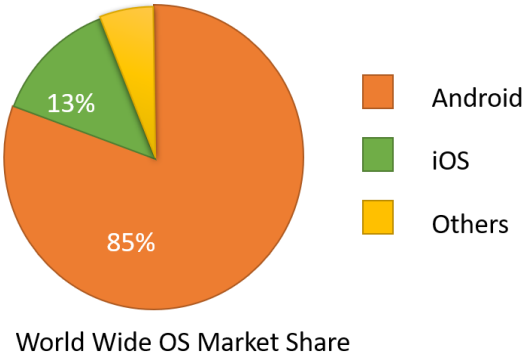

Figure 1: Worldwide Market Share of various Mobile OS

3. DEEP LEARNING

Deep Learning is energized by the function of the brain, utilizes Artificial Neural Network and thus are also known as DNN [1]. Neural networks are built with neuron nodes (just like human brain) connected like a web. Here, deep mentions the number of hidden layers in the neural network. Deep learning model can analyze unstructured data. Deep Learning techniques need more powerful hardware than Machine Learning [2]. Machine Learning is mostly applied in predictive programs, email spam identifiers and the like. Deep Learning is used in fields like facial recognition, music streaming services and self-driving cars.

Among various deep learning models, few are supervised learning-based models and others are unsupervised learning-based models. Supervised models are Multilayer Perceptron (MLP), CNN and Recurrent Neural Network (RNN). MLP model is also known by the name Neural network or Artificial Neural Network (ANN) model. A basic neural network with one hidden layer is shallow neural-network. More than one hidden layer is DNN [3].

Unsupervised learning models are Self-organizing maps, Boltzmann machines and Autoencoder. There are some variations of Autoencoder. They are: Sparse, Denoising, Contractive and Stacked Autoencoder.

Further, depending on the features which are used to categorize an application, Machine learning/Deep learning can be arranged as: Static analysis, Dynamic analysis and Hybrid analysis.

3.1 Static Analysis

Static Analysis analyzes compiled file without executing it and therefore can be used on small devices like smart phones which have constrained memory. Static analysis does not need any specific requirement to be fulfilled, does not need to set up any environment and therefore can give results in very less time [4]. This makes static analysis a good choice. Researchers give preference to static analysis as compared to other methods.

3.2 Dynamic Analysis

Dynamic analysis analyzes the runtime behavior of applications like Application Programming Interface (API) calls. During runtime, an execution path is followed during which algorithm analyzes the application. Therefore, dynamic analysis must aim to maximize code coverage [5]. Dynamic analysis is resilient to code obfuscation and is the only way to detect new unknown malwares. This is the reason that researchers have begun to use dynamic analysis for their research. However, limitation of dynamic analysis is that it cannot be used on memory constrained devices [6]. Secondly, if sensitive section of the code is not covered during execution, analysis will miss the malware [7].

3.3 Hybrid Analysis

Hybrid analysis is the combination of static and dynamic analysis where it takes the results from static analysis and uses them for improving the efficiency of dynamic analysis [6]. Though, it has same limitation as of dynamic analysis: it consumes ample resources and is time consuming.

4. LITERATURE SURVEY

4.1 Deep Neural Network based system

DNN is the neural network with more than one hidden layer [8]. It's the network of neurons with input features and hyperparameters and use activation function to make the transformation non-linear [9]. Though, DNN are quite powerful, they bear a problem of vanishing gradients. Figure 2 shows a Deep Neural network with 2 hidden layers.

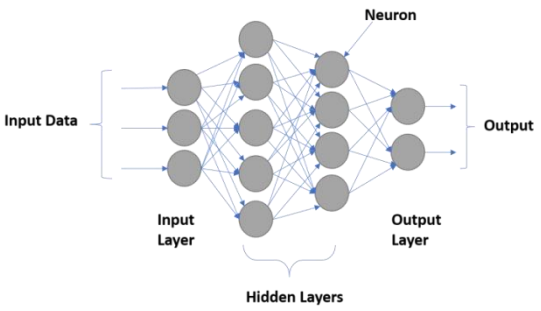

Figure 2. Deep Neural Network

Karbab et al. [1], developed a framework, MalDozer. The author used raw sequences from API calls. Framework was developed using neural networks and first layer of the network is

convolution layer. The reason why author has used convolution layer is that they automatically find out the pattern in raw method calls. MalDozer can detect unknown malware samples and can also attribute them to their family. Moreover, because of small amount of preprocessing, it can be deployed on small devices. The framework worked on three datasets: Malgenome, Drebin & MalDozer with achieved accuracy of 99.18%, 98% & 85% respectively.

Li et al. [2], proposed a detection engine with the name, DeepDetector using DNN. Not limited to the classification of malware or benign, detection engine classifies the apps among various malware families using softmax activation in the output layer. Authors proved their detection engine to be better than machine learning methods with accuracy of 97.16%. This paper used 5000 malware samples and 125,000 benign samples.

In [3], authors have implemented variations of deep learning neural networks and used 7 different static features including string feature, method opcode & method API features, shared lib function opcode feature, permission, component and environmental features. Virusshare and Malgenome datasets are used with 20,000 malware samples from virusshare, 1260 from malgenome and 20,000 benign samples from google play store. Accuracy with DNN when all features are used is 94%.

Sirisha P et al. [4], used DNN and with 331 permission features, they were successful in detecting malware with the accuracy of more than 85%. Permission features were extracted using androguard package in python. Dataset was taken from Droid bench and Google play store with 398 apk files.

Naway et al. [5], made a survey of latest deep learning techniques that are used for android malware detection. Authors surveyed papers based on static, dynamic and hybrid analysis. It was concluded that static analysis was dominant but gradually other analysis methods are being adopted by researchers.

Alzaylaee et al. [6], proposed a deep learning framework for Android malware detection via dynamic analysis and used stateful input generation for enhanced code coverage. The paper worked with real devices instead of emulators. Experiments were performed on 31,125 apps and accuracy achieved was 97.8% with only dynamic features and 99.6% with both static and dynamic features. The paper also does comparison with 7 machine learning classifiers.

Sandeep HR [7] used deep learning techniques for android malware detection through static analysis. Author implemented fully-connected deep learning model with a dataset of 331 features. Accuracy achieved was 94.64% and validated using Random Forest classifier.

4.2 Convolutional Neural Network based system

In Convolutional Neural network, neurons are not deeply connected [10]. Unlike other models, CNN can itself find relevant features in the data without human intervention. CNN is used mostly for image classification but can be used for non-image data as well [11].

In Nannan et al. [8], presented a framework Andro_MD implemented using CNN, actually both sequential and parallel CNN models for android malware detection.

Sequential model is said to perform better. Permissions features are used belonging to 7 different categories. Total 34,570 features are used from 21,000 apps. The paper compared its performance from similar works and traditional machine learning classifiers and claimed to perform better. Achieved accuracy was 96.25%.

Zhang et al. [9], proposed a system named DeepClassifyDroid. It is based on CNN using static analysis. The paper used set of static features for implementation where accuracy achieved was 78.4% when only permission features are used and 97.4% when combination of all features is used. The paper made comparison with 3 other machine learning approaches (KNN, Linear SVM & Naïve Bayes) and claims to be 10 times faster than Linear-SVM. Datasets used is Drebin with 5546 malware samples.

Huang et al. [10], propose a system known as R2-D2 which is based on CNN and works on bytecode from Android archive file. Bytecode is converted to RGB color code to get a image for input to CNN. The system can analyze tons of real time data at much faster pace. Moreover, as CNN is used, so human labor is reduced while feature extraction process during training phase. Datasets are taken from various sources with 2 million samples including malicious and benign android apps and malware detection accuracy was 98.4225%.

Zou et al. [11], static analysis is implemented using CNN. The model processes raw bytecode and bytecode is robust to the obfuscation techniques. One limitation of the ByteDroid though is that it restrains the bytecode sequence to 1500KB during training stage. Detection accuracy was 92.17% and datasets used are from VirusShare, FalDroid, Android PRAGuard and Kang et al.

Wang et al. [12], proposed a framework that used static analysis and implemented CNN algorithm of deep learning. The paper used static features and extracted 1003 of them. Dataset is taken from Google play store that included 5000 malicious and 5000 benign samples and accuracy attained was

99.68%. Contribution of the paper is the new malware detection method for android OS.

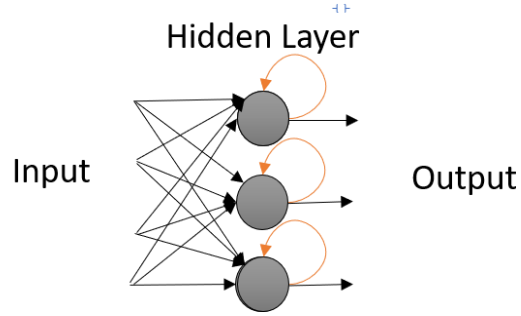

Figure 3: Recurrent Neural Network

4.3 Recurrent Neural Network based system.

RNN is mostly used for time series analysis and natural language processing [15]. Training examples equate to one another. Example is the analysis of stock market prices.
Current prices are correlated to the prices of previous months or years. They are important for future predictions.

Figure 2 shows the design of RNN in which hidden layer has two roles to play: to give output and to feedback to itself. This is in relation with the correlation characteristic [16].

Nauman et al. [13], used more than one neural network models to perform Android malware analysis. Authors used fully-connected neural network, CNN, RNN & Long Short-Term Memory (LSTM), Autoencoders and DBN. Highest accuracy achieved is by LSTM of 93.6% and CNN with accuracy of 89.5%. Authors have compared all deep learning algorithms with Bayesian Machine Learning and they were able to overpower the results of NN methods at the accuracy of 99.8% but at the cost of lost coverage. Few apps were difficult to classify even with Bayesian method.
Zhang et al. [14], implemented RNN for android malware detection and used both static and dynamic features.

Table 1 encapsulates the work of all the papers discussed here to make a comparison based on the algorithms they used, key concept, features extracted for the purpose of implementation, accuracy, their contributions and limitations.

Table 1: Comparison of Deep Learning based Classifiers

Ref. No.	Year	Key Concept	Features	Dataset	Algo-rithm	Accuracy	Contribution	Limitation
[1]	2017	Static Analysis	Raw sequences of API method calls	Malgenome, Derbin, MalDozer	DNN	99.18%	Put forward an automatic feature extraction technique.	Not Strong against dynamic code loading & Obfuscation.
[2]	2018	Static Analysis	Static features	125,00 benign & 5000 malicious samples.	DNN	97.16%	Present DeepDetector engine that has high precision and low FPR.	Only static features are used.
[3]	2018	Static Analysis	7 static features	VirusShare, Malgenome, Google Play store	DL	Highest average accuracy of 98%	Implemented many models of Deep Learning	Dynamic features are not included.
[4]	2019	Static Analysis	Permissions (331 features)	Droidbench, Android play store	DNN	85%	Successfully detected malware in real-time android apk files.	Only static features are used.
[5]	-	Static, dynamic & hybrid analysis	-	-	DL	-	Survey of Deep Learning methods.	-
[6]	2019	Dynamic Analysis	420 static & dynamic features	31,125 apps including 11,505 malwares	DL	99.6%	DL-Droid DL system for malware detection using real devices.	-
[7]	2019	Static Analysis	331 static features	VirusShare	DL	94.64%	Can also detect Android malware app name & version packages.	Dynamic features are not included.
[8]	2018	Static analysis	Permission features	21,000 apps	CNN	96.25%	Presented Andro_MD framework & Verified	Dynamic features are not

							that requested & used permissions have better performances.	included.
[9]	2018	Static Analysis	Combinatio bn of static features	Drebin	CNN	97.4%	Presented a system for Android Malware detection DeepClassifyDroid.	Only static features are used.
[10]	2018	Static Analysis	Bytecode	2 million benign & malicious android apps.	CNN	98.4225%	Proposed R2-D2 system to process and analyze tons of real-time data faster than before. Also Reduced human labour.	Complex system
[11]	2019	Static Analysis	Bytecode	VirusShare, FalDroid, Android PRAGuard, Kang et al.	CNN	92.17%	No manual feature extraction.	Based on static analysis.
[12]	2019	Static analysis	1003 Static features	Google play store.	CNN	99.68%	Provided new android malware detection method	Only static features are used.
[13]	2017	Static Analysis	Static features	Drebin, VirusShare	DL	Acc.98.9 % F1 98.6%	Implemented many NN algorithms and Bayesian ML.	Implemented only static features.
[14]	2018	Static & dynamic analysis	Static & dynamic features	Total 1986 apks collected, benign samples from Google play store	RNN	-	Compared performance with Logistic Regression & SVM and also with deep learning (MLP) and proved better.	-

5. CONCLUSION

The revolution in smart phone usage has captivated the attention of malware authors. Android users are more at risk owing to Android's open nature. Major concern is the detection of Android malware and this paper made a survey of existing deep learning techniques and frameworks like MalDozer, Deep-Detector, R2-D2 and ByteDroid. Most of the papers have implemented DNN and CNN. Highest accuracy achieved is 99.68% by a CNN algorithm. Most of the papers have used static analysis. However, integrating two or more techniques can improve accuracy and detection rate which is our major concern.

REFERENCES

[1] ElMouatez Billah Karbab, Mourad Debbabi, Abdelouahid Derhab, andDjedjiga Mouheb. "Android Malware Detection using Deep learning on API method sequences." Published December 2017.

[2] Dongfang Li, Yibo Xue, and Zhaoguo Wang. "DeepDetector: Android Malware Detection using Deep Neural Network." International Conference on Advances in Computing and CommunicationEngineering, France 22-23 June 2018.

[3] TaeGuen Kim, BooJoong Kang, Mina Rho, Sakir Sezer and Eul Gyu Im. "A Multimodal Deep Learning Method for Android Malware Detection using Various Features." IEEE Transactions on Information ensics and Security (Volume: 14, Issue: 3, March 2019).

[4] Sirisha P, Kamala Priya B, Aditya Kunal K, and Anuradha T. "Detection of Permission Driven Malware in Android Using Deep Learning Techniques." 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA) 2019. DOI: 10.1109/ICECA.2019.8821811.

[5] Abdelmonim Naway, and Yuancheng LI. "Use of Deep Learning in Android Malware Detection." International Journal of Computer Science and Mobile Computing December-2018, Vol.7 Issue.12, pg. 42-58.

[6] Mohammed K. Alzaylaee, Suleiman Y. Yerima, and Sakir Sezer. "DL- Droid: Deep learning based android malware detection using real devices." Computers & Security November 2019, DOI: 10.1016/j.cose.2019.101663.

[7] Sandeep HR. "Static Analysis of Android Malware Detection using Deep Learning." Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS 2019) IEEE XplorePart Number: CFP19K34-ART; ISBN: 978-1-5386-8113-8.

[8] Nannan Xiea, Xiaoqiang Di, Xing Wang, and Jianping Zhao. "Andro MD: Android Malware Detection based on Convolutional Neural Networks." International Journal of PerformabilityEngineering vol. 14, no. 3, March 2018, pp. 547-558 DOI: 10.23940/ijpe.18.03p15.547558.

[9] Yi Zhang, Yuexiang Yang, and Xiaolei Wang. “A Novel Android Malware Detection Approach Based on Convolutional NeuralNetwork.” ICCSP 2018: Proceedings of the 2nd InternationalConference on Cryptography, Security and Privacy. 144-149. 10.1145/3199478.3199492.

[10] TonTon Hsien-De Huang, and Hung-Yu Kao. “R2-D2: ColoR-inspired Convolutional NeuRal Network (CNN)-based Android Malware Detections.” IEEE International Conference on Big Data (Big Data) 2018. DOI: 10.1109/BigData.2018.8622324.

[11] Kewen Zou, Pengfei Liu, Weiping Wang, Xi Luo, and Haodong Wang. “ByteDroid: Android Malware Detection Using Deep Learning on Bytecode Sequences.” October 2019. 10.1007/978-981- 15-3418-8_12.

[12] Zhiqiang Wang, Geifi Li, Yaoing Chi, Jianyi Zhang, Tao Yang, and Qixu Liu. “Android Malware Detection based on Convolutional Neural Networks.” CSAE 2019: Proceedings of the 3rd International Conference on Computer Science and Application Engineering. 1-6. 10.1145/3331453.3361306.

[13] Mohammad Nauman, Tamleek Ali Tanveer, Sohail Khan, Imam Abdulrahman, and Toqeer Ali Syed. “Deep neural architectures for large scale android malware analysis.” Cluster Computing 21(3):1-20 March 2018. DOI: 10.1007/s10586-017-0944-y.

[14] Jianming Zhang, Futai Zou, and Junru Zhu. “Android Malware Detection Based on Deep Learning.” IEEE 4th International Conference on Computer and Communications,Chengdu, China, 2018, pp. 2190-2194, DOI: 10.1109/CompComm.2018.87810.