A New Malware Classification Framework Based On Deep Learning Algorithms

Seminar Report

Submitted in partial fulfillment of the requirements for

the award of degree of

Bachelor of Technology

in

Computer Science and Engineering

of

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

Chandana Govindan

MAC19CS019

KNOWLEDGE IS POWER

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MAR ATHANASIUS COLLEGE OF ENGINEERING, KOTHAMANGALAM

KERALA

December 2022

A New Malware Classification Framework Based On Deep Learning Algorithms

Seminar Report

Submitted in partial fulfillment of the requirements for

the award of degree of

Bachelor of Technology

in

Computer Science and Engineering

of

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

Chandana Govindan

MAC19CS019

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MAR ATHANASIUS COLLEGE OF ENGINEERING, KOTHAMANGALAM

KERALA

December 2022

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MAR ATHANASIUS COLLEGE OF ENGINEERING

KOTHAMANGALAM

CERTIFICATE

This is to certify that the report entitled **A New Malware Classification Framework Based On Deep Learning Algorithms** *submitted by* **Chandana Govindan (MAC19CS019)**, *towards partial fulfillment of the requirement for the award of Degree of Bachelor of Technology in Computer Science and Engineering from APJ Abdul Kalam Technological University for December 2022 is a bonafide record of the seminar carried out by him under our supervision and guidance.*

....	....	....
Dr.Elizabeth Isaac	**Prof.Pristy Paul T**	**Prof. Joby George**
Seminar Guide	***Seminar Coordinator***	***Head of the Department***

Date: Dept. Seal

ACKNOWLEDGMENT

I express my sincere gratitude and thanks to Dr. Bos Mathew, Principal and Prof. Joby George, Head of the Department for providing the necessary facilities and their encouragement and support.

I owe special thanks to the project guide Dr. Elizabeth Isaac and the project coordinator Prof. Pristy Paul T for their corrections, suggestions and sincere efforts to co-ordinate the seminarunder a tight schedule.

I express my sincere my thanks to staff members in the Department of Computer Science and Engineering who have taken sincere efforts in guiding and correcting me in conducting this seminar

Finally, I would like to acknowledge the heartfelt efforts, comments, criticisms, co-operation and tremendous support given to me by my dear friends during the preparation of the seminar and also during the presentation without which this work would have been all the more difficult to accomplish.

Abstract

Nowadays, there are more crimes committed in cyberspace. Malicious software (malware) is frequently used by cyber criminals to destroy victim machine or steal the important data. Detection and classification of these malwares is a challenging task, traditional machine learning (ML) algorithms are not efficient for malware detection and classification. So, a deep learning (DL) approach with a hybrid model of two pretrained networks ResNet-50 and AlexNet can be used for malware classification. In the first step, the malware sample is converted into equivalent grayscale image using vision-based technique. After that feature extraction is done using the pretrained networks AlexNet and ResNet-50. Then malware classification is done by passing the feature vector generated by the networks through three fully connected layers and a softmax layer. The evaluation shows that the proposed method can classify malware with high accuracy than any traditional methods.

Contents

List of Figures

List of Abbreviations

ReLU	Rectified Linear Unit
CNN	Convolutional Neural Network
GPU	Graphics Processing Unit
DL	Deep Learning

Chapter 1

Introduction

Modern computer technology and the Internet have made life more simple and easy for people. Nowadays, everything can be done online, including social networking. interaction, exchange of money, etc. These improvements all entice online users.encouraging criminals to carry out their crimes online rather than in real world. Recent findings in science and business claim that Cyberattacks have an economic cost of trillions of dollars. Malware is a common tool used by cybercriminals to launch cyber-attacks. Any software known as malware engages in unauthorised and suspicious actions on the computers of its victims. Malware can be divided into different categories, including viruses, Trojan horse, rootkit, worm, ransomware, etc.

In order to safeguard computer systems, it is crucial to promptly identify malware as soon as it infiltrates them. The process of scrutinizing suspicious files and determining whether they are benign or malware is known as malware detection. After confirming that a file is indeed malware, the next step is to classify it into a specific category or family, which is referred to as malware classification.

The model employed data from various malware datasets such as Microsoft BIG 2015, Malimg, and Malevis. Initially, grayscale images of malware samples were generated and fed into the Deep Learning system. Once the image acquisition process was complete, the proposed hybrid architecture's convolution layers were utilized to extract the malware's high-level features. The model was then trained under supervision. To create the hybrid model, the suggested approach merged several comprehensive deep-learning models that relied on a transfer-learning approach. The model employed a Rectified Linear Unit (ReLU) function and multiple hidden layers.

1.1 Motivation

To improve the detection rate, various techniques and technologies such as data science, machine learning, heuristics, cloud computing, big data, and block chain are utilized in malware detection procedures. By leveraging these methods and technologies, there are multiple strategies that can be employed for malware detection. These approaches mainly include heuristic-based detection, model checking, and signature, behavior, and behavior analysis. Signature-based detection is effective in identifying known and related malware variants. However, it cannot detect malware that has not yet been encountered. Although behavior-based, heuristic-based, and model checking-based detection methods can detect some unknown malware components, they are not sufficient for identifying complex malware variants that are difficult to detect.

Deep learning is being used as a novel strategy to get around the limits of the existing malware detection and classification techniques. While deep learning has been extensively employed in a variety of fields, including image processing, computer vision, and the recognition of human actions, it has not been frequently applied in cybersecurity, notably in the detection of malware. Artificial neural networks (ANNs), which have numerous hidden layers and learn from examples, are used in deep learning, a type of artificial intelligence. Several deep learning architectures, including as deep neural networks (DNN), deep belief networks (DBN), recurrent neural networks (RNN), and convolutional neural networks, have recently been designed in order to improve the performance of the model (CNN). The suggested approach makes use of CNN architecture.

1.2 Objective

Malaware is a software designed to damage and destroy computers and computer system. Malware detection and classification is necessory in order to protect our system. Here proposing a deep learning approach with a hybrid model of two pretrained networks ResNet-50 and AlexNet.

Chapter 2

Literature Review

Malware identification and classification is a drawn-out procedure. These phases employ a range of methodologies and technologies. Malware must first be evaluated with certain software in order to be detected.The outcomes are recorded, and features are either manually or automatically retrieved. Here, data mining methods are utilised to obtain useful attributes[2]. Following that, the retrieved features are chosen in accordance with to a set of requirements. Finally, machine learning is used to train a subset of features.To differentiate malware, use algorithms or rule-based learning techniques. [2],[4].

With vision-based approaches, there are two main techniques for feature extraction and malware visualisation. By expressing malware binaries as images, the first method does away with the requirement for further feature extraction tools like Sandbox, Disassembly, or API Monitor. This method transforms the malware binary first into an 8-bit vector and then into a 2D array[1].

Artificial neural networks (ANNs) are the foundation of the discipline of artificial intelligence known as deep learning, which learns from examples. Although it has been extensively used in a number of fields, including voice control, driverless cars, and image processing, it has not been properly applied to the detection and classification of malware. Although the deep learning-based approach decreases the feature dimension and gives great performance, it is vulnerable to evasion attacks[2]. Moreover, creating hidden layers takes effort, and adding more hidden layers improves performance just somewhat. Deep learning is not widely used in the detection and categorization of malware, thus more academic study is needed to determine how effective it is.

The threat posed by malware to modern computers is growing quickly. In order to produce new malware variants and avoid detection by current malware detection technologies, malware authors frequently integrate advanced features like code obfuscations. Even when the classifier is trained with known variants from the same family, classifying unknown malware variants with comparable characteristics into their respective families is a considerable difficulty.

Chapter 3

Methodology

3.1 Architecture

This section outlines the deep learning-based malware classication framework . For malware classification, this system offers a mixed deep neural network architecture. The suggested system's technique, as shown in Figure 3.1, consists of three basic components. First off, a number of comprehensive datasets are used to acquire the malware data. Second, pre-trained networks are used to extract both low-level and high-level malware traits. Finally, supervised learning is used to carry out the training phase of our deep neural network design.

Malware visualisation and model overview are the two main divisions within this section. The malware visualisation section presents grayscale images of malware variants in binary file format. The recommended approaches for the malware classification framework are fully explained in the model overview section.

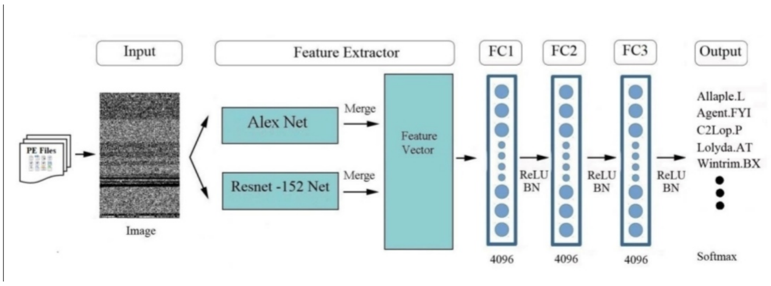


Figure 3.1: Proposed malware classification methodology.

In order to solve a number of difficult situations that arise during the classification process, such as time limits and high dataset dimensions, transfer learning algorithms have been investigated. The feature extraction procedure in the transfer learning approach makes use of pre-trained networks. The classification process is completed by using a general classifier, such as a support vector machine or softmax. This approach is specifically designed to cater to the proposed architecture, ensuring it can handle the aforementioned challenging conditions effectively.

3.2 Malware Visualization

Converting binary code into images can be achieved through various methods, and in this study, the approach of visualizing executable malware binary files was utilized. The primary objective was to represent binary files as grayscale images. Figure 3.2 shows how malware binary files are transformed into grayscale pictures. The malware binary file is initially read as an 8-bit unsigned integer vector. Then, each component's binary value is changed to its matching decimal value. A 2D matrix is created from the resulting decimal vector, which is then transformed into a grayscale image. Based on the size of the malware binary file, the 2D matrix's dimensions are chosen. The size of the malware binary file determines the matrix's width and height. Examples of image frames used for malware visualisation from various malware families are shown in Figure 3.3.

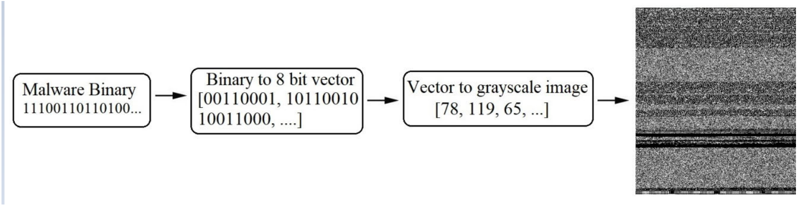
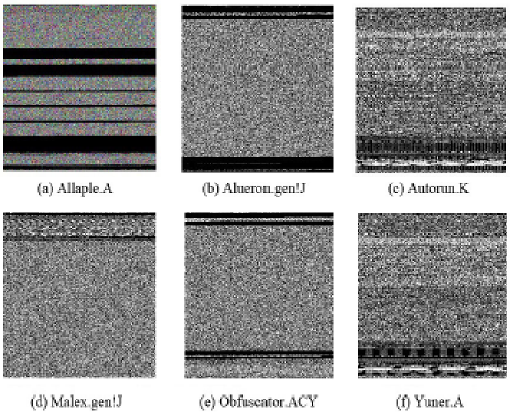


Figure 3.2: Overview of malware visualization process



(a) Allaple.A (b) Alueron.gen!J (c) Autorun.K

(d) Malex.gen!J (e) Obfuscator.ACY (f) Yuner.A

Figure 3.3: Malware grayscale images from different malware families

3.3 Model Overview

With the use of a hybrid deep neural network architecture, the proposed model provides an optimised framework for categorising malware. The suggested framework is divided into four stages, as shown in Figure 2: data gathering, deep neural network architecture design, training, and evaluation. The first datasets from which malware data is gathered are Malimg, Microsoft BIG 2015, and Malevis. In the following section, the specifics of these datasets are described. Two pre-processing phases are part of the proposed deep neural network architecture. Predicting an appropriate DL architecture for use in malware classification is the first step. A hybrid module [54] could offer higher overall precision, according to preliminary tests. In order to do this, pre-trained ResNet-50 and AlexNet architectures were combined to form a hybrid module.

To address various challenges faced during the classification process, like limited time and large dataset size, transfer learning techniques were explored. In this approach, pre-trained networks were utilized for feature extraction, followed by using a broad classifier like the softmax or support vector machine for the final classification step. This approach was specifically adapted for the proposed architecture to effectively handle the aforementioned challenging conditions.

The suggested model combines Resnet and AlexNet architectures by creating a feature vector using an equal weighting procedure and using it to train a system to have a high accuracy rate. There are three steps in the procedure. First, the ImageNet dataset is used to pre-train the Resnet and AlexNet architectures. Then, a feature vector with 4096 dimensions is created by combining the features collected from the two architectures. The pre-trained ResNet-50 and AlexNet architecture features are 2048 dimensional, respectively. After obtaining the combined feature vector, normalisation is produced by passing it through the softmax layer and fully connected layers. The fully connected layers are made up of 4096 nodes to increase the suggested network's capacity for learning, while the softmax layer generates 57 outputs that correspond to the 57 malware classifications.

3.4 Flowchart

The proposed model offers an improved framework for classifying malware, using an architecture for a hybrid deep neural network. There are four steps in this framework's methodology, collecting malware data, designing the network architecture, training the model, and evaluating its performance. Figure 5 presents a detailed flowchart of the system, This has three sections and four stages. The pre-trained networks serve as feature extractors during the pre-training phase. Fully connected learning layers are included in the training stage, and a softmax classifier serves as the top layer for classification.

Using a model that has been trained on a vast amount of data for our problem is called transfer learning. As a result, we exclusively teach them through model optimization. The model will train quickly, which is a plus for us.A stored network that has previously undergone training on a sizable dataset, generally for a sizable image-classification task, is referred to as a pre-trained model. Either apply transfer learning to adapt the pretrained model to a specific task, or use the model as is.

The idea behind transfer learning for image classification is that if a model is trained on a sizable enough dataset with adequate generality, it will be able to represent the visual world as a whole. By training a big model on a big dataset, you can then use these learnt

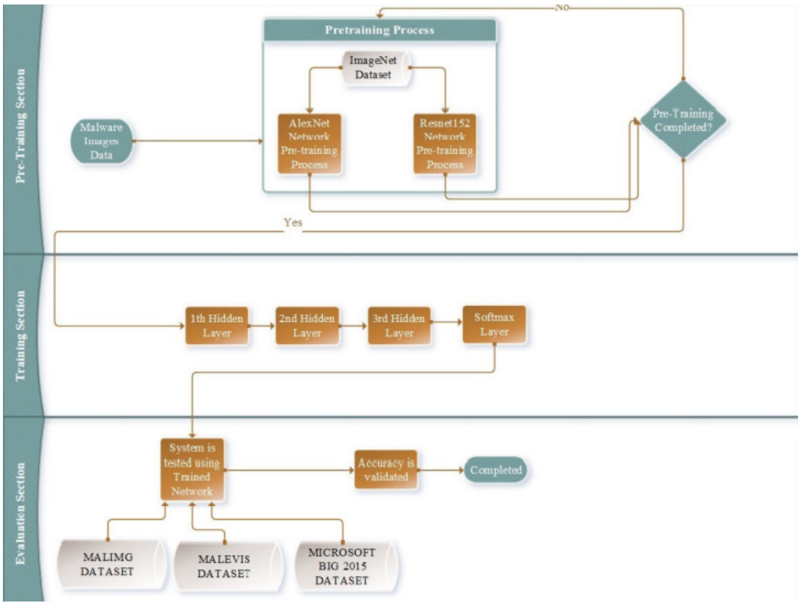


Figure 3.4: Flowchart of proposed deep learning architecture for malware classification

feature maps to your benefit instead of having to start from scratch.

Chapter 4

Pretrained Networks

4.1 ResNet-50

A convolutional neural network known as ResNet-50, which is shown in Figure 6, has won the ILSVRC 2015 and COCO 2015 competitions. 50 layers make up the deep neural network that it is. Five convolutional blocks make up the network model, and they use convolution layers of 1 * 1, 3 * 3, and 1 * 1. The ResNet-50 network also has a fully connected layer, a softmax layer, and two pooling procedures. There are 25.6 million parameters in this architecture altogether.

Figure 4.1 shows the ResNet design, which consists of a convolutional layer with a kernel size of 77 and 64 distinct kernels, all of which have a stride of size 2. Max pooling with a stride size of two follows. A 11, 64 kernel appears in the next convolution, then

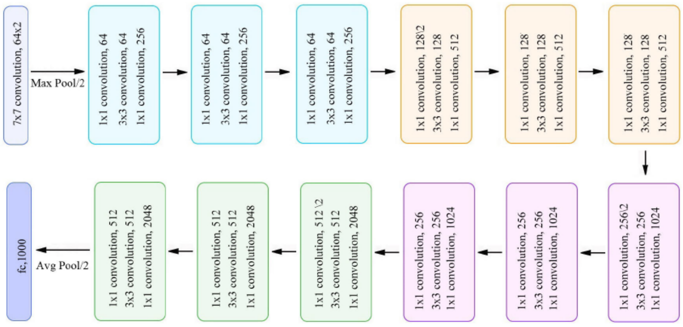


Figure 4.1: ResNet-50 Architecture

a 33, 64 kernel, and eventually an 11, 256 kernel. In this stage, these three layers are repeated three times to create nine layers. A 33, 128 kernel comes next, then an 11, 512 kernel, and finally an 11, 128 kernel. Twelve layers are created by repeating this method four times. The following kernels are 11, 256, 33, 256, and 11, 1024, followed by two additional kernels. Six times through this process, a total of 18 layers are created. The following kernels are 11, 512, 33, 512, and 1*1, 2048, each repeated three times to create nine layers. One layer is then formed by average pooling, a fully connected layer with 1000 nodes, a softmax function, and so on. The activation functions and maximum/average pooling layers are not counted. Hence, the deep convolutional network in the ResNet design includes a total of 50 layers.

4.1.1 Skip Connection

Deep neural networks are advantageous in that they are able to learn intricate functions more efficiently compared to their shallow counterparts. However, during the training of deep neural networks, the model's performance tends to decrease as the depth of the architecture increases. One of the reasons behind this issue may be overfitting, but in this case, it is not the only reason. The deeper neural network with 56 layers has a higher training error than the shallower one with 20 layers, as shown in the figure below. Therefore, the deeper model does not perform as well as the shallower one. Another possible reason for this performance degradation could be the vanishing or exploding gradient problems. Nevertheless, the creators of ResNet argued that by utilizing Batch Normalization and appropriate initialization of weights through normalization, the gradients are ensured to have healthy norms.

He et al. introduced Residual Networks in 2015 to address image classification challenges. ResNets use matrix addition to transfer information from the initial layers to deeper layers, without requiring extra parameters. The output of the previous layer is added to the next layer via a skip connection. A residual block with skip connection is represented as follows:

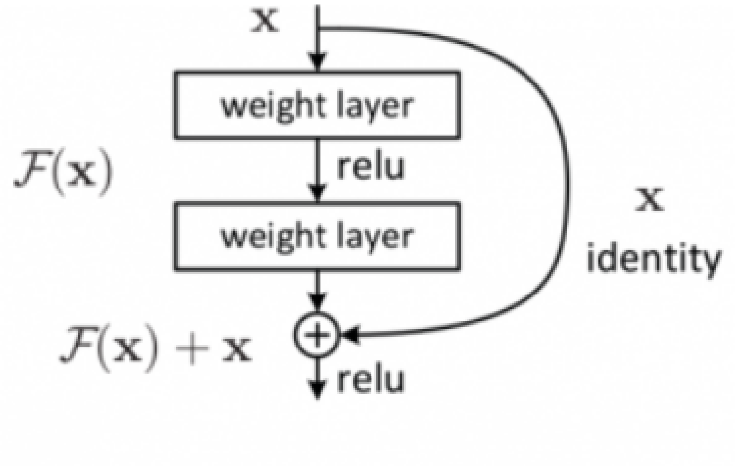

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Figure 4.2: Skip Connection

ResNets have a deep layer representation that makes them versatile for solving multiple tasks, beyond image classification. Pretrained weights from ResNet can be applied to tasks like image segmentation, keypoint detection, and object detection. As a result, ResNet has become a highly influential architecture in the deep learning community.

4.2 AlexNet

The ImageNet Large Scale Visual Recognition Challenge introduced the popular convolutional neural network AlexNet. It has an 8-layer, straightforward design, with the first five being convolutional layers and the latter three being fully linked layers. To aid in the learning and classification operations, the network also has two normalisation layers, three pooling layers, seven ReLU layers, and a softmax layer. Figure 4.2 provides an illustration of the architecture.

AlexNet demonstrates that ReLU non-linearity can be used to train deep CNNs more efficiently than saturating activation functions like Tanh or Sigmoid. The figure illustrated below exhibits that AlexNet can achieve a training error rate of 25 percent with the aid of ReLUs (solid curve).

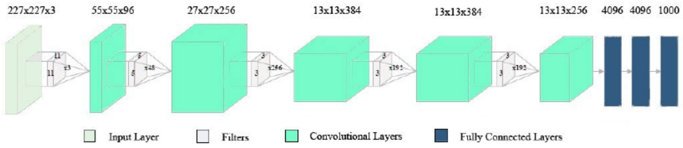


Figure 4.3: AlexNet Architecture

The first convolutional neural network to use GPUs to improve performance was AlexNet. Five convolutional layers, three max-pooling layers, two normalisation layers, two fully connected layers, and one softmax layer make up its architecture. Each convolutional layer uses ReLU as the nonlinear activation function. Pooling layers are used to perform max pooling. The input size is fixed by the presence of completely connected layers. The padding makes the input size, which is frequently stated as 224x224x3, into 227x227x3. AlexNet includes 60 million parameters in total.

Chapter 5

Conclusion and Future Work

The difficulty of accurately detecting malware variants still poses a severe danger to the field of cyber security despite extensive research on malware detection and classification due to code obfuscation and packing techniques. This work suggests a new deep learning architecture that uses a hybrid technique to efficiently detect malware variants.Several pre-trained networks using the transfer learning method are used in the approach to extract features from a sizable dataset of malware data. A supervised learning technique is then used in the training phase to train the deep neural network design.

5.1 Future Work

By utilizing this model, a small portion of malware instances could not be accurately categorized due to their utilization of sophisticated code obfuscation methods and resemblance to other types of malware. Future research will focus on designing a detection mechanism that can specifically identify and classify malware that uses such obfuscation techniques.

References

[1] L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, “Malware images: Visualization and automatic classication,” in Proc. 8th Int. Symp. Visualizat. Cyber Secur. (VizSec), 2011, pp. 17.

[2] Ö. Aslan and R. Samet, “A comprehensive review on malware detection approaches,” IEEE Access, vol. 8, pp. 62496271, Jan. 2020.

[3] R. Gupta and S. P. Agarwal, “A comparative study of cyber threats in emerging economies,” Globus, Int. J. Manage. IT, vol. 8, no. 2, pp. 2428, 2017.

[4] R. Komatwar and M. Kokare, “A survey on malware detection and classi- cation,” J. Appl. Secur. Res., pp. 131, Aug. 2020

[5] W. Huang and J. W. Stokes, “MtNet: A multi-task neural network for dynamic malware classication,” in Proc. Int. Conf. Detection Intrusions Malware, Vulnerability Assessment. Cham, Switzerland: Springer, 2016, pp. 399418.

[6] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2015, pp. 1–9

[7] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–778

[8] A. Singh, A. Handa, N. Kumar, and S. K. Shukla, “Malware classification using image representation,” in Proc. Int. Symp. Cyber Secur. Cryptogr. Mach. Learn. Cham, Switzerland: Springer, Jun. 2019, pp. 75–92.

[9] Z. Cui, F. Xue, X. Cai, Y. Cao, G.-G. Wang, and J. Chen, “Detection of malicious code variants based on deep learning,” IEEE Trans. Ind. Informat., vol. 14, no. 7, pp. 3187–3196, Jul. 2018, doi: 10.1109/tii.2018.2822680.

[10] Y. Ye, D. Wang, T. Li, and D. Ye, “IMDS: Intelligent malware detection system,” in Proc. 13th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD), 2007, pp. 1043–1047

PAPER NAME

MALWARE.pdf

WORD COUNT	CHARACTER COUNT
3653 Words	**20894 Characters**
PAGE COUNT	FILE SIZE
26 Pages	**1.4MB**
SUBMISSION DATE	REPORT DATE
Mar 16, 2023 11:11 AM GMT+5:30	**Mar 16, 2023 11:11 AM GMT+5:30**

● **33% Overall Similarity**

The combined total of all matches, including overlapping sources, for each database.

- 26% Internet database
- Crossref database
- 0% Submitted Works database
- 26% Publications database
- Crossref Posted Content database